THE COMPUTER-AIDED EVALUATION AND SYNTHESIS OF A DATA COMMUNICATIONS NETWORK

A THESIS
SUBMITTED IN PARTIAL FULFILMENT
OF THE REQUIREMENTS FOR THE DEGREE
OF
MASTER OF SCIENCE IN COMPUTER SCIENCE
IN THE
UNIVERSITY OF CANTERBURY
BY

CLINTON ANDREW LOGAN

University of Canterbury 1991

To my Parents,
Frances and Desmond

*"I'd be glad to improve myself,"
he said, "but I don't know how to go
about it. What shall I do ?"*

From SAGGY BAGGY ELEPHANT by K. & B. Jackson,
©1947 by Western Publishing Company, Inc.

ABSTRACT

With most terrestrial telecommunication networks experiencing growth the need for powerful computer design tools is becoming mandatory. Such tools facilitate the quick and accurate quantification of many complex technical and economical interactions enabling planners to control the evolution of their networks.

This thesis focuses on several issues surrounding the computer-aided design of a wide area data communications network. Three main topics are addressed: the application of interactive computer graphics to network design tools; the inherent shortcomings of several contemporary design methods; and the application of the tools developed during this study for the evaluation of an existing wide area network.

Network Designers Workshop (NDW), the computer planning tool presented in this thesis has been developed to address some of the main inadequacies found in current day design tools. NDW utilizes high resolution graphics to provide the designer with a highly interactive framework for the rapid prototyping of communication networks.

In addition, NDW's network synthesis methodology emphasises the importance of adopting an integrated approach to network design by enabling the planner to find a minimum cost solution through a series of iterative designs.

The architecture and facilities of a modern packet switching network are also examined with a special focus on the mechanisms available for the collection of the essential performance data needed for the evaluation and design stage.

The final section of this thesis concentrates on the application of the design tools presented in this study for the evaluation and cost driven optimization of a multimillion dollar packet switching network. Finally the impact of nodal cost and access network tariff structures on the optimum cost topology are illustrated.

ACKNOWLEDGEMENTS

I have looked forward to writing this page for some time. I can finally acknowledge the help that people have given me during my three year study.

First I want to thank **Ray Hunt** for the support and continuous encouragement he provided during the course of this study.

My thanks go to **Robert Davidson** (TNI Ltd) for the valuable discussions we have had over the last two years— pity we did not get the email link up sooner.

I am grateful to **Netway Communications** for giving me the opportunity to play with something "real" and **Lincoln Phelps**, the network engineer, for answering a multitude of questions — toward the end he must have been grateful we were separated by several hundred miles of telephone wire.

My **Father** and **Grandmother** for proof-reading the manuscript, which resulted in a slightly less than infinite number of small improvements and clarifications.

Special thanks are due to **Datacom** for the financial assistance of the Battersby Scholarship.

I offer my appreciation for the help and cooperation I have received from the technicians within the department, especially **Tony**, who sold me coffee at vastly inflated rates when I needed it most.

Thanks to **Apple Computer** for making the ultimate thesis writing machine: The Macintosh IIci, which made the typing and retyping of the 281,921 characters constituting the final manuscript a pleasure instead of a chore. I wonder how that will sound in ten years time!

The greatest thanks of all however, must go to **Nicola** for giving me the help, love, and warmth that kept me going throughout my six years at University. I could not have done it without her.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

chapter one

INTRODUCTION

Can your network accommodate an increased load of twice the present level? — if so then how much more can it handle? As the load increases which components will saturate first? How reliable is the network and what are the consequences of this link failing? Is the current network guaranteeing adequate performance for its users all the time? Is there a more cost-effective way of engineering the network?

These are but a few of the hundreds of questions network managers must concern themselves with today. Surprisingly, many network operators address such problems on a "patch as needed" basis resulting in a network simply evolving with little attention being given to its global structure. Design decisions are often made without any in-depth analysis of the network, since they are commonly addressing localized expansions. Such a process seldom results in a cost-effective structure.

With most terrestrial telecommunications networks experiencing growth the need for powerful design tools is becoming ever more apparent. Network managers must analyze a vast range of technical and economic issues and be able to quantify the impacts of the complex interactions between them. Multiple tradeoffs must be evaluated for different network expansion or design alternatives to guarantee successful migrations to future cost-effective configurations.

OUTLINE OF THE STUDY

This thesis is concerned primarily with the development and subsequent use of computer based tools for the analysis and design of packet based communication networks. This study illustrates how a structured design strategy is essential for the successful management of any network system and offers the potential to provide significant cost savings.

MOTIVATION

The NET-X packet switching network is a privately owned system that provides a data communications service to a large number of customers within New Zealand. NET-X is one of the two main packet switching networks in New Zealand and has grown substantially over the last three years. In the light of this rapid expansion, the evaluation of the network's current design and performance was of critical importance.

After an initial two week investigation of the NET-X environment it was discovered that there was a complete lack of any analysis or planning tools. The network's management system did have comprehensive monitoring facilities that collected information on each component, however there were no true analysis tools for evaluating and summarising network performance over an extended period of time.

In addition to the development of the data collection and analysis tools some form of network design tool was needed. A literature review was conducted to gain an understanding of the current state of play in the network design tool arena. It was discovered that although there were a tremendous number and variety of design packages available, the vast majority were lacking in several key areas.

Thus, the primary motivation for this study was this industry need for improved computer-aided design tools that would enable the planning and dimensioning of wide area data communication networks.

OBJECTIVES

This study was initiated with two primary goals in mind —

• First, to highlight deficiencies in contemporary design tools and to develop a "network designer's workshop" of design capabilities that would support the planning of complex communication networks. Such a tool would provide a visual *exploratorium* for the rapid evaluation and synthesis of cost-effective network designs. This approach can be realised with current computer hardware; high speed workstations with large high-resolution displays, dedicated processors, and sizable amounts of virtual memory to support the levels of interactivity required by such a tool.

Network Designers Workshop (NDW), the computer-aided planning tool presented in this thesis, addresses several shortcomings present in existing tools.

• The second goal was to illustrate the utility of NDW in the evaluation of network designs by assessing the current NET-X structure and to orientate its configuration toward a target network optimized with respect to the cost tradeoffs and performance goals.

APPROACH

In order to achieve these goals the following multi-phased approach was adopted:

• *Data collection and analysis* — The development of an accurate picture of the current network state is a key step in any design evaluation project. Several data collection tools were developed to enable the off-line analysis of the traffic statistics maintained by the network management system. These analysis tools provided the network operators with a previously unavailable window into the operational state of their network.

- *Design tool development* — The second and longest phase involved the design and development of Network Designers Workshop, a graphically orientated network design environment.
- *Network evaluation and design* — The third and final phase involved the coupling of NDW with the data obtained from the workload characterization step to evaluate the current NET-X design.

THESIS CONTRIBUTIONS

The primary contributions of this thesis are (1) NDW a network design tool prototype; (2) Insight into the current state and design deficiencies of a communications network; and (3) the proposal of an alternative NET-X structure that exhibits an 18% increase in traffic carrying capacity with an annual cost saving of $84,427.

Secondary contributions of this research include some insight gained into several aspects of the network design problem. The optimisation approach adopted by NDW illustrates the need for integrated design methodologies; Several inherent shortcomings of a classical design heuristic are highlighted and improvements to it proposed; Finally the impact of several design parameters on the optimum cost topology are illustrated.

THESIS STRUCTURE

This thesis is organized as follows —

Chapter Two provides a brief introduction to communication networks. It covers the main concepts that arise within this work and includes a brief look at several topological structures.

Chapters Three, Four and *Five* constitute Section II of this thesis and collectively provide an overview of communication network design. *Chapter Three* introduces the network design problem, including a formal description and finally formulates the global design strategy adopted by this study. *Chapter Four* presents the design methodologies used in this study and highlights some of their inherent deficiencies. The field of Computer-Aided Network Design (CAND) is discussed in *Chapter Five.* Several 'state-of the-art' tools are presented.

Section III presents Network Designers Workshop (NDW) the design tool developed during the course of this study. *Chapter Six* introduces NDW and provides an overview of the tool — including its design philosophy, architecture and functional features. Several implementation issues surrounding NDW are presented in *Chapter Seven,* including a closer look at its internal design algorithms, and verification of the tool based on network measurement data.

The analysis of the NET-X packet switching network is covered in Section IV. A description of the Hughes Integrated Packet Network (IPN) environment is provided by *Chapter Eight.* The current NET-X configuration is examined, including its topology, subscribers, and network monitoring facilities. Chapter eight concludes with a look at the data collection and analysis tools developed during this study.

Chapter Nine discusses how the measurement tools were employed to evaluate several aspects of NET-X's operational state. The results of this analysis are presented along with several issues surrounding the workload characterization phase.

Chapter Ten concentrates on the cost driven optimization of the NET-X structure using the traffic measurements and NDW. The impact of several design parameters on the optimized structure are also illustrated in the latter sections of this chapter.

Chapter Eleven, the final chapter, provides a summation of the contributions made by this study, including the network improvements that have occurred as a result of this work.

SECTION I

COMMUNICATION NETWORKS: AN OVERVIEW

Let no one enter who does not know geometry
Inscription on Plato's door

c h a p t e r   t w o

NETWORKING CONCEPTS

Whenever two or more entities are connected to exchange information, a network is formed. This chapter serves as an introduction to networks and discusses the main concepts that arise in this study.

NETWORK CLASSIFICATION

Broadly speaking, networks can be classified by several attributes. These attributes can be grouped according to the network classification tree detailed in figure 2.1.

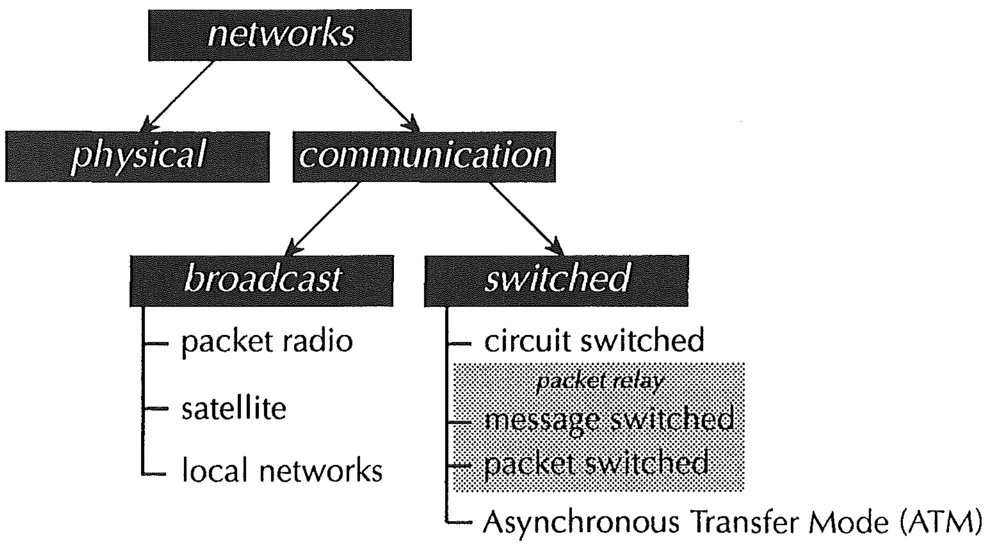

Figure 2.1 Network classification hierarchy

NETWORK TYPE

Networks can be divided into two broad types: Physical networks and Communication networks. *Physical networks* are designed for the transport of physical items such as water, boxes etc. *Communication networks* are designed for the transport of discrete units of information.

The techniques used for the design and modelling of physical and communication networks are very similar, however the nomenclature used is quite different [27].

GEOGRAPHICAL COVERAGE

Networks range in size from small Local Area Networks (LAN) which connect several computers within a single building, to medium sized systems that interconnect computers within a metropolitan area or city (MAN), to large geographically distributed Wide Area Networks (WAN) covering entire countries and even spanning the globe.

TRANSMISSION TECHNIQUE

Data communication networks can also be classified according to the type of transmission technique they employ. Two types are in common use — *Switched communication* and *Broadcast communication* networks.

Switched communication networks

Networks that provide communication between all users fall into the category of switched networks. Three classes of switched networks exist:

- *Circuit switched networks* — provide services by establishing a dedicated communication path between the two users for the duration of the call (e.g. Public Service Telephone Network).

- *Packet relay networks* — forward logical units of data from source to destination. Packet relay networks discard the need for a dedicated channel between the source and destination. Rather, if a subscriber wants to send data, it is passed from node to node until the destination is reached. At each node the data units are queued along with other data and then forwarded. *Message switched* and *packet switched* networks represent the two fundamental types of packet relay networks in operation today. Packet switching is very similar to message switching, the principle difference being the length of the data units.

- *Cell relay networks* — have recently emerged to accommodate the requirements for fast, large scale data transfer. One term that describes cell relay technologies is Asynchronous Transfer Mode (ATM). ATM uses small packets (known as cells) which are switched through the network. Cells have a fixed size of 48 bytes with a 5 byte header. The use of fixed length cells allows the simplification of the transfer protocol and switching hardware and therefore greatly increasing throughput. ATM is the primary mode of data transfer for broadband ISDN.

This thesis deals with the analysis, design and modelling of packet relay networks, of which thousands are in operation today.

Broadcast networks

Broadcasting offers the main alternative to switched communication networks. In a broadcast network a source will transmit information to a large number of subscribers over a shared transmission medium. The message is received by all subscribers, but only the intended recipient will accept and process the message. Broadcasting is the main transmission technique employed by local area networks.

Several design tools for broadcast networks can be found in the literature [3, 70], however they are beyond the scope of this study.

NETWORK COMPONENTS

An abstract model of a wide area data communications network, including its major components, is portrayed in figure 2.2. This model is characteristic of the thousands of packet based networks in operation today.

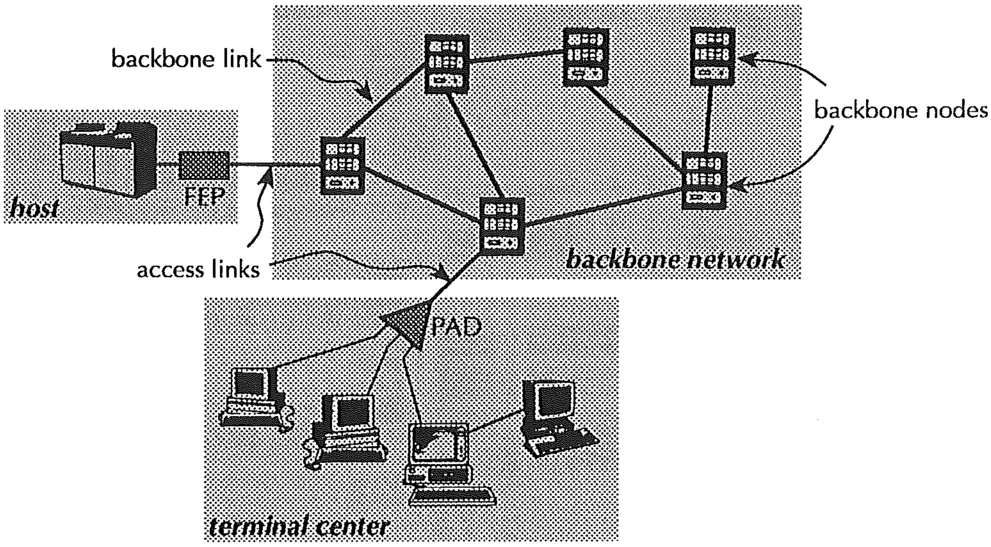

Figure 2.2 Generalised network structure

The primary function of any network is to connect a number of geographically distributed subscribers enabling them to communicate and share resources.

Terminal centres and *Host* computers act as sources and sinks for data traffic and are collectively termed *subscribers* within this thesis. A terminal centre consists of a cluster of terminals connected by some form of terminal concentrator (i.e. PAD, MUX etc). Host computers are generally directly connected by way of a Front End Processor (FEP) to a backbone node. A subscriber is connected to a backbone node by way of a dedicated or dial-up *access link*.

Communication networks can be viewed as a two level hierarchical structure—the lower level consisting of the *subscriber access network* and the upper level representing the *backbone network*. The backbone network consists of a system of switches or nodes interconnected by *backbone links* and is primarily responsible for the transportation of the subscriber traffic to its destination.

Backbone *nodes* or *switches* are specialised computers that are used to connect two or more links together. Their primary function is to process incoming data and route it over the outgoing links. Nodes are also responsible for handling various other tasks such as flow control, buffering and protocol enforcement.

NETWORK TOPOLOGY

One of the most important design considerations is how the network components should be connected. The network topology describes the interconnection patterns of these components.

The choice of topology is to some extent dependent on the user application. If the network is expected to share data between many pairs of widely dispersed subscribers then the least cost topology will be an irregular meshed structure (like those in figures 2.4, 2.5 and 2.6). If however, the data is shared between a central site (which could be a node, host, or terminal concentrator) then the optimal structure can be found using methods developed for centralised topologies [30].

This section attempts to classify some of the more notable topological structures. Some elementary topologies that have proven useful for data communication networks include:

- *Distributed mesh topology* — can be either fully or partially connected. In a fully connected distributed mesh every node is connected to every other node. Fully connected meshes offer high reliability, although the cost of the system grows with the square of the number of nodes.
- *Minimal spanning tree or multidrop topology* — consists of a master node or root with branches radiating from it. These branches may in turn have sub-branches. MST/MD topologies are used when all communication is bound for a single destination, and are frequently employed for subscriber access topologies.
- *Star Topology* — all nodes are connected via point to point links to a central site.
- *Bus topology* — nodes are connected by way of short drops to a shared broadcast link. Bus topologies are employed by local area broadcast networks.
- *Ring topology* — all nodes are connected in series along a closed loop. Ring topologies are normally used for LANs and MANs.

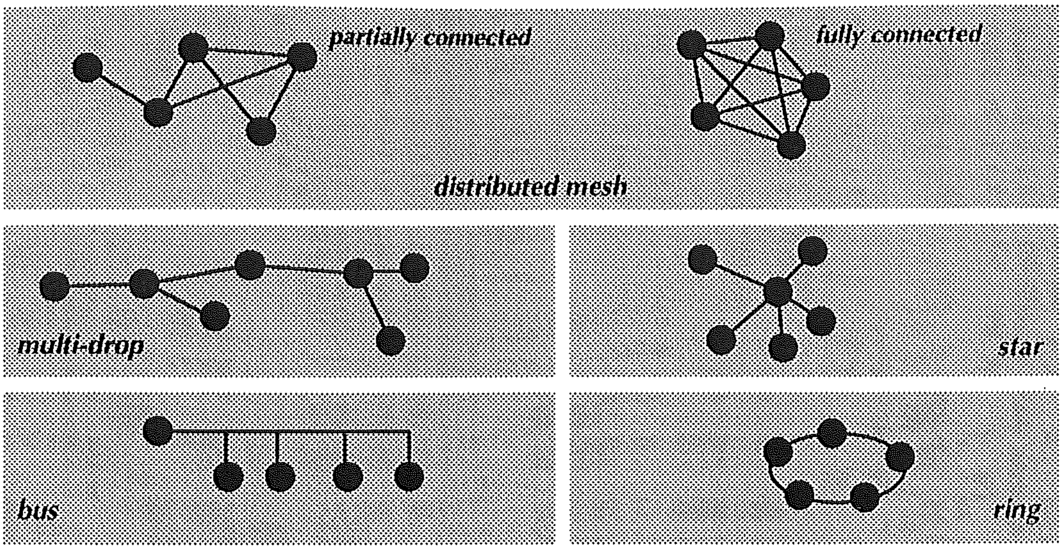

Figure 2.3 Common network topologies

Other structures can be obtained by combining the topologies in figure 2.3. Sharma [82] provides a good discussion on the topologies covered in this section.

DISTRIBUTED MESH TOPOLOGIES

This thesis concentrates on the topological design of a wide area network where demands can arise between any two nodes. Distributed mesh structures offer a cost effective means of connecting those nodes. The economical design of such topologies however, requires the detailed knowledge of traffic flows among all subscribers [82].

Distributed mesh structures can be divided into roughly two generic categories:

Multicentre multistar (MCMS) topology

A *multicenter multistar* (MCMS) topology is a hierarchical structure that consists of several nodes that individually serve a subset of the subscriber population (see figures 2.4 and 2.5). The subscribers are connected to the closest backbone switch using a star configuration.

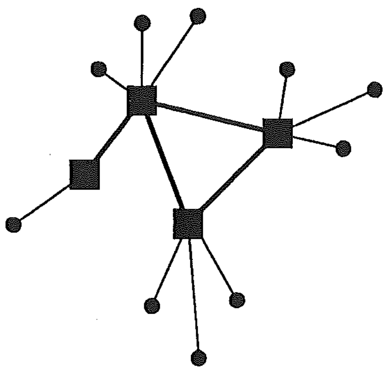

Figure 2.4
Two Level, MCMS Topology

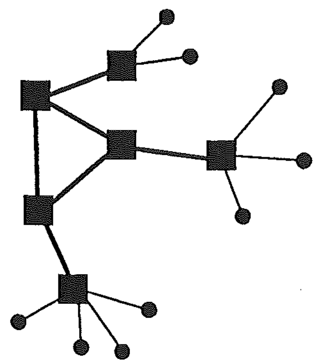

Figure 2.5
Three Level, MCMS Topology

The NET-X packet switching network discussed in Section IV employs a two level MCMS topology.

Multicenter multidrop (MCMD) topology

A *multicentre multidrop* (MCMD) topology is a hierarchical network that can be used to serve many subscribers over a large geographical area.

Figure 2.6 illustrates a typical MCMD structure with four backbone switches. The subscribers are interconnected using shared multidrop access links.

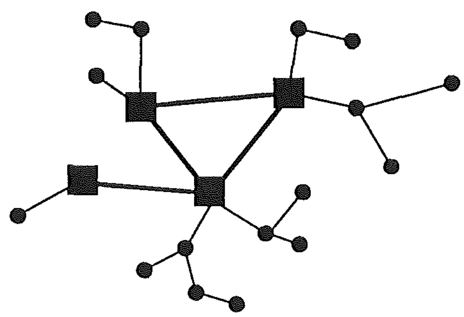

Figure 2.6 Two Level, MCMD Topology.

Large complex communication networks typically consist of a combination of the topological structures discussed in this section.

TRAFFIC ATTRIBUTES

Communication networks are characterized by flows of discrete data units from one subscriber to another. Several attributes that characterize packet based network traffic are enumerated in the following paragraphs.

Traffic unit

Although many measures of traffic unit exist (e.g. bits, bytes, frames, messages, transactions etc), the *packet* represents the fundamental traffic unit for network design. A data packet consists of a variable length user information segment encapsulated by a fixed length header.

Traffic intensity

Traffic intensity is a term used to describe the data traffic load on a network component. Traffic intensity can be defined as the traffic arrival rate per unit time and is denoted by the symbol λ. Traffic intensity is normally expressed in *Packets Per Second* (PPS).

Traffic delay

The traffic delay imposed by a system can be measured in three primary ways:

• *Transit delay* — is the time that elapses from when a packet enters a component until it exits. The transit delay imposed by nodes and links are characterized by the sum of their queueing and service times.

• *Average network delay* — is the average time it takes for a data packet to travel from its source to destination nodes. The designer must be aware of the dangers associated with sizing networks using this measure. An acceptable average network delay does not always guarantee acceptable end-to-end delays for all node pairs [77].

• *Response time* — The user response time can be defined as the time that elapses from when the operator sitting at a terminal completes the last action associated with a command until the first character of the host's response is received. Any definition of response time that is applicable to network design should eliminate host turn around delays, which effectively reduces it to the network transit delay.

Methods for calculating traffic delays are discussed in Chapter four.

Traffic throughput

An efficient design ensures a high traffic throughput for a given network cost and traffic delay. The throughput of a system is defined as its total traffic carrying capacity per unit time.

The following three measures of throughput are of interest to the network designer:

• *Communication link throughput* — Transmission link capacities are rated in bits per second (bps). Recommendation X.135 defines the link throughput to be the number of user *data* bits successfully transferred in one direction over that connection per unit time. To determine the throughput of a link, communications protocol overhead must be taken into account. Thus the maximum achievable throughput of a link is always less than its transmission rate.

The maximum theoretical throughput[1] of an X.25 link is given by [37]:

$$\textit{link throughput} = \frac{S}{[8(8+D)]}\ PPS \tag{2.1}$$

where S = link speed in bps and D = user data packet length in bytes.

• *Nodal throughput* — Whilst communication link capacities are well defined and the corresponding throughputs can be calculated, the calculation of nodal

[1] Assuming no flow control and piggybacked frame acknowledgments.

throughput is much less certain. In virtually all cases, throughput calculations are dependent on the manufacturers own capacity ratings. Nodal capacity ratings are specified in terms of Packets Per Second (PPS), however manufacturers normally do not supply any information on how general protocol overhead should be taken into account. The difference between the *actual* data packet processing capability and the manufacturers rated throughput of a node is dependent on the internal implementation and can vary quite considerably.

- *Network throughput* — For a given *average network delay*, the total network throughput represents the rate at which the network can transport traffic. Network throughput provides a measure of the network's traffic carrying capacity and a convenient mechanism for comparing designs.

SECTION II

DATA COMMUNICATION NETWORK DESIGN

When a lot of remedies are suggested for a disease,
that means it can't be cured
Anton Chekhov
The Cherry Orchard, 1904

chapter three

THE NETWORK DESIGN PROBLEM

The design of data communication networks in its general form has prompted a tremendous amount of research interest, resulting in a wide range of approaches to the problem. Some of these approaches simplify the real world case, so established optimization techniques can be applied [30], whereas others artificially restrict the solution space in order to decrease the search time [31]. The most effective methods to date employ heuristic procedures to arrive at a *close to optimum* structure in an acceptable time.

This chapter provides an introduction to the Network Design Problem (NDP) and is organised into three major sections — (1) *The design problem* introduces the NDP in its general form; (2) *The problem formulation* provides a more rigourous definition of the NDP; and (3) *The design strategy*, details the global design strategy adopted by this study.

THE DESIGN PROBLEM

The predominant goal in the design and maintenance of a communications network is to provide a cost-effective[1] structure that meets several key criteria. Those criteria are generally accepted to be:

• *High throughput* — The network should have the capacity to absorb traffic fluctuations and serve the growing requirements of subscribers without any degradation of service quality.

• *Short transmission delays* — Traffic should be transferred from source to destination in the minimum possible time, thus ensuring acceptable response time for its users.

• *High levels of reliability* — Network subscribers should be able to communicate even if network components fail.

• *High levels of extendibility* — The topology should facilitate the graceful introduction of new nodes and backbone links without the need for significant redesign of the existing structure.

---

[1] The term *cost-effective* is used in place of minimal cost for two primary reasons — First, the inherent nature of network design (to date) ensures that a minimal (or optimum) cost structure is unattainable [10]; the best we can hope for is a "close to optimum" design. Second, the minimal cost structure is not always the prime requirement of the network design problem.

The design of a cost-effective distributed data communications network represents a formidable combinatorial problem. Such problems epitomise the characteristics and general difficulties of the NDP as the number of possible solutions is sufficiently vast to prohibit the exhaustive search for an answer.

For example, having selected the geographic locations for *n* backbone nodes, we can then connect them using *m* backbone links. The total number of possible designs to consider is determined by [26]:

$$\frac{[n(n-1)/2]!}{[n(n-1)/2-m]!\ m!} \qquad (3.1)$$

NET-X has values of n = 16 and m = 22, so there exist 6.313×10^{23} possible topologies. Given Network Designers Workshop can evaluate a topology in 100ms, an exhaustive search for the locally optimum topology[2] would take 2029 trillion years![3]. To make things worse, Frank's analysis ignores the fact that the determination of the globally optimum topology also depends on the selection of the node locations — If there are *s* subscribers in a network, then the optimal number of backbone nodes *q* will lie somewhere in the range $1 \leq q \geq s$.

To determine the optimum network configuration successive values of *n* must be tried until $n = q$ is reached. Each iteration will involve the evaluation of every possible set of inter-nodal connections given by formula 3.1!

This phenomenon is termed a *combinatorial explosion*. To combat problems of this nature, a class of algorithms called heuristics need to be applied.

Fortunately several design heuristics have been developed to significantly reduce the time needed to find a solution. The tradeoff however, is that the heuristic no longer guarantees to find the 'best' solution. A discussion of network design design heuristics can be found in the following chapter.

PROBLEM FORMULATION

In its most general sense, the network design problem corresponds to many different choices in performance measures, design variables and constraints [30]. Many formulations can be found in the literature; the following has been adopted:

DESIGN INPUT

The input to the network design problem consists of the following:

---

[2] Chapter four discusses the difference between locally and globally optimum topologies.

[3] About 405,000 times the age of the Earth — clearly ruling out this approach!

- *The number and location of the network subscribers* — The positions of the network subscribers (x_i,y_i) i= 1,2..s can be described using any valid set of two dimensional coordinates (e.g. the V and H coordinate system, latitude and longitude etc).

- *The traffic requirement between subscriber pairs* — the traffic requirements are represented by a *traffic matrix* of the form $TM = (TR_{src,dest})$ src,dest = 1,2..s where $TR_{src,dest}$ represents the traffic intensity originating at src and terminating at dest.

- *The transmission link cost tariffs* — The transmission link alternatives (e.g. ADS, DDS, Megalink etc.) and their corresponding cost structures.

- *The hardware cost profiles* — The hardware alternatives available for the final solution and an accurate mechanism for evaluating their cost.

- *Message profiles* — Knowledge of the average length and distributions of all data and control messages flowing within the network.

DESIGN OUTPUT

Output from the network design problem will provide the following:

- *The optimum number of backbone nodes* — The optimum number q where $1 \leq q \geq s$. s = number of network subscribers.

- *The optimum locations for the backbone nodes* — the positions of the q backbone nodes that minimise the tradeoff between subscriber access and backbone link costs.

- *The backbone connection topology* — the most cost-effective interconnection pattern joining the nodes using the most suitable link types (e.g. DDS, ADS etc.) and capacities.

- *Node dimensioning* — the optimum number of processing modules[4] as a function of the nodal traffic handling requirements.

- *Miscellaneous empirical information* — network costs, utilization, delays etc.

DESIGN CONSTRAINTS

The solution to the network design problem is to be achieved subject to many different constraints including:

- The maximum average packet delay through the network;
- The total network throughput at the maximum network delay;
- The maximum average delay on any network link;

---

[4] Assuming the node has a modular architecture (refer figure 8.3).

- Network reliability[5].

DESIGN OBJECTIVE

The primary objective of the network design problem is to find the least cost topology that satisfies a specified set of design constraints. The NDP however is not necessarily limited to finding the least cost structure, other objectives may include:

- The maximum throughput topology [19].
- The maximum reliability topology [34, 49].
- The minimum mean delay topology [33].

THE DESIGN STRATEGY

The successful management of any network is a continuous process. The goal of such a process is to produce and maintain a close to minimum-cost network configuration whilst still satisfying the performance requirements of its subscribers. Because any reasonably sized network is a dynamic entity constantly undergoing change, the need for its constant re-evaluation is paramount to maintaining the aforementioned goal.

The operation and management of a network can be broken down into three key activities. Figure 3.1 identifies the three phases that make up the network *design cycle*.

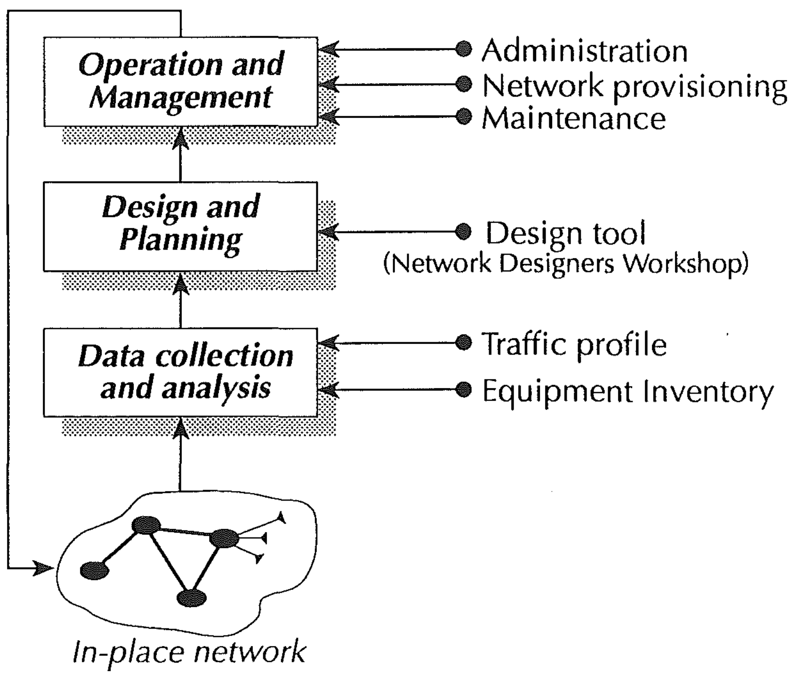

Figure 3.1 The network design cycle

[5] e.g. two-connected — a node must be connected to at least two other nodes.

This process will involve the coordination of several departments and logically starts at the data collection and analysis phase. The cycle then proceeds (using the data from the first step) through the design and planning stage and on to the operation and management stage. The final phase uses the insight gained by the design stage to improve on the existing network configuration.

This cyclic process of network management is representative of the planning strategy adopted by many organisations today [10, 91, 97].

The following sections discuss the three phases in more detail.

DATA COLLECTION AND ANALYSIS PHASE

The data collection and analysis of the existing network represents a fundamentally important step of the design process.

The network management facilities that exist on modern systems provide detailed statistics on network activity (i.e. traffic volumes, Community Of Interest (COI) data[6], nodal utilizations etc). This information is vital for evaluating the network's ability to meet service and utilization objectives and is also used to highlight potential trouble spots where corrective planning action may be necessary. The data gained by this phase provides vital input to the design and planning phase.

Section IV discusses the data collection and analysis phase of this study.

DESIGN AND PLANNING PHASE

The goal of the design and planning phase is to develop a cost-effective network configuration that provides the desired level of performance. This new configuration will either provide a basis for network expansion or verification of the present network. *"The key to accurate network engineering is the ability to feed network statistics and COI data, as well as forecast growth, into an interactive network design program containing the characteristics of the network components"* [10].

The designer, armed with the data gathered in the first phase, uses a computer-aided design tool to postulate "what if" scenarios. The designer can experiment by moving nodes and links and quickly observe the resulting shifts in traffic distributions. Components can be deleted to simulate failure conditions and the loads due to alternate routing can be identified to ensure sufficient capacity on the secondary traffic paths. Good tools can be used to synthesize new low cost topologies or to plan the expansion of a current network. A tool that allows the designer to experiment with such hypothetical scenarios is essential for the successful management of any network.

---

[6] Who is talking to whom and how much.

The use of Network Designers Workshop (NDW) for the execution of the design and planning phase is covered in Chapter ten.

OPERATION AND MANAGEMENT PHASE

The third phase of the design cycle is responsible for the acceptance and execution of the results gained from the design phase. This step includes tasks such as equipment acquisition, installation, maintenance and general administration of the network.

The insight gained by this study has resulted in several improvements[7] to the existing NET-X topology.

[7] The modifications are discussed in the second section of Chapter eleven.

chapter four

DESIGN METHODOLOGIES

Data communication networks of any sort, are the product of practical engineering design. These networks exhibit a tremendous variation in structure and represent a considerable financial investment by their owners.

The desire to configure these networks so costs are minimized, and yet still meet the design criteria, is an entirely reasonable one. A cost decrease of just 5% on a network costing $160,000 per month represents a monthly saving of $8000. This saving would easily justify a considerable amount of effort in the design stage.

This chapter provides an overview and discussion of the main design methodologies available to the network engineer with an emphasis on those used in this study. Network design methods can be split into three key design activities; performance prediction, topological design and network cost analysis.

PERFORMANCE PREDICTION

The performance evaluation of network components, typically backbone nodes, switches, PADs etc., forms an essential part of the network design process.

Unfortunately the analysis of measurement data[1] can only provide insight into the state of an existing network. To evaluate the performance of planned networks predictive techniques are necessary.

PERFORMANCE MODELLING

The task of performance modelling involves the representation of a real world system in terms of some abstract quantity. Network models must be constructed in order to reflect the performance characteristics between proposed network configurations.

Performance evaluation methods were greatly enhanced by Kleinrock's work in the delay analysis of packet switching networks [53, 54] and over the last two decades a tremendous amount of research effort has been dedicated to the performance evaluation of computer networks.

Three primary methods for forecasting the performance of data communication networks exist, namely:

---

[1] The analysis of measurements on the NET-X environment are detailed in Chapter nine.

- *Physical models* — require the availability of the actual system under study. Tests are performed using artificial loads to obtain performance statistics. Physical models represent the most expensive and time consuming approach.
- *Simulation models* — involve the software emulation of the network components and processes. A simulation model describes the *dynamic* behaviour of the system. Network objects (i.e. packets) and their corresponding sequences of activity (i.e. packet flows) are directly represented through software event procedures. Simulation models permit the representation of a system at an essentially unlimited level of detail and generally are employed for the more detailed examination of system behaviour (e.g. protocol performance evaluation, buffer utilization etc.)

One of the primary drawbacks of simulation is the cost involved in constructing and running the model. Simulations represent a significant programming task and require considerable amounts of computer resources to execute.

- *Analytical models* — are a mathematical representation of the real world system. Analytical models are a group of solution techniques that describe the functional relationship between system parameters (e.g. traffic arrival rates, processing capacities etc.) and a chosen performance criterion (e.g. utilizations, delays etc.). Because of their applicability to this study, analytical methods are discussed in further detail in the following section.

Several tutorials exist which bear on the subject of network performance modelling; the more notable ones include Reiser [74], Tobagi et al. [93], and Inose [46].

Selecting a modelling technique

There are several considerations that must be made before a modelling technique is adopted. The selection is partially based on the required results and partially on the available tools[2].

One of the key considerations is the desired level of modelling accuracy. Both simulation and analytical approaches involve the abstract representation of the system under study. In general, analytical methods require a much higher level of abstraction than simulation. Simulations are not bound by the same assumptions that are necessary to make analytical expressions mathematically tractable, consequently more accurate results can be expected.

As previously mentioned, this additional gain in accuracy is made at the expense of a more costly model to develop and run. This extra cost may not always be justified or desirable. In the case of Network Designers Workshop, the objective to provide high levels of interaction ruled out the use of simulation techniques.

---

[2] These tools include measurement facilities, modelling skills, and programming languages.

Kobayashi points out that *"An analytic model should be sought wherever possible, since it can evaluate the performance with minimal efforts and costs over a wide range of choices in the system parameters and configurations"* [55].

Simulation models have primarily been used in cases where analytical models have not or cannot be derived. Simulation, despite its performance drawbacks, still represents a powerful class of tools for the design and performance analysis of computer networks [4, 45].

This study has adopted analytical modelling techniques, since they provide a fast[3] and proven method for the performance prediction of packet relay networks. The analytical models implemented by NDW are discussed in the following section.

ANALYTICAL PERFORMANCE MODELS

Among the number of mathematical disciplines that are pertinent to analytical modelling, queueing theory plays the most important role. We can visualize a computer network as a multiple resource system, where the resources are the network nodes, transmission links, PADs etc. Network traffic demand services from these resources, and as a consequence, most performance problems are related to the queueing delays experienced from the contention for these resources.

The performance models adopted by this thesis are based on the work done by Kleinrock [54], and Ng [69]. These models are based on an *open* network of queues. They are open because all sources generate new packets and all arrivals are assumed to be accepted by the network without any flow control. Open queueing models are accurate for light to medium loads where flow control mechanisms have little or no effect on performance. When loads increase however, the open model overestimates network delays.

In practice most packet relay networks implement sliding window protocols for flow control. The use of *closed* queueing network models for the analysis of flow controlled networks has been advocated by Schwartz [79], Reiser [75] and others. Although closed queueing models enable more accurate modelling of flow controlled networks, they suffer from the drawback of requiring a greatly increased amount of computational time, space, and complexity for calculating network performance measures[4].

Thus, the following open queueing models have been adopted by this study and are dependent on the following assumptions [54]:

---

[3] In terms of programmer time required to implement the model and computer resources needed to run the model.

[4] Lam and Hsieh [98] have developed a heuristic that significantly reduces the time needed to run these models.

- The traffic arrivals form a Poisson process;
- The packet length[5] is exponentially distributed[6];
- Each queue works on a First Come First Served (FCFS) discipline;
- The arrival processes are independent at every switch;
- Every switch has infinite buffer space.

Transmission link model

A full-duplex communication link is responsible for the transmission of data packets. In such a process the transmission (or service) time will be linearly proportional to the packet length. Thus a full-duplex link can be successfully modelled by the M/M/1 queue detailed in figure 4.1.

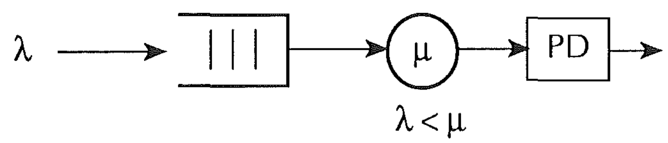

Figure 4.1 Transmission link M/M/1 queueing model

The total delay (T_{link}) imposed by a transmission link can be determined by:

$$T_{link} = \frac{1}{\mu C - \lambda} + PD \qquad (4.1)$$

Where

$\frac{1}{\mu}$ = Average packet size (in bits).

C = Capacity of the link (in bps).

λ = Mean packet arrival rate for a link (in PPS).

PD = Link propagation delay (in sec).

Backbone node model

The increasing demand for packet switching networks has spurred the development of a second generation of packet switching nodes which are characterised by their highly modular architectures[7]. Nodes consist of several

[5] Equivalent to the service process.

[6] Figure 9.8 and its subsequent discussion in Chapter nine illustrate the correlation between this assumption and actual network measurements.

[7] A more detailed description of modern nodal architectures can be found in Chapter eight.

Packet Switching Clusters (PSC) interconnected by a high-speed internal bus or ring (refer to figure 8.3).

When a packet arrives at a PSC it is buffered, processed, and passed on. Each PSC contains buffer space and processing units and acts independently of the others. Thus there exists several parallel packet queues within each node.

It is reasonable to expect that the PSC service time should be independent of the packet length (i.e. deterministic), because the operation of each PSC involves the inspection of a fixed-length packet header and the re-routing of that packet [24]. A PSC can therefore be effectively modelled as an M/D/1 queue. A node consisting of m PSCs and a total traffic arrival rate of λ_t is modelled by the m-M/D/1 queueing structure in figure 4.2.

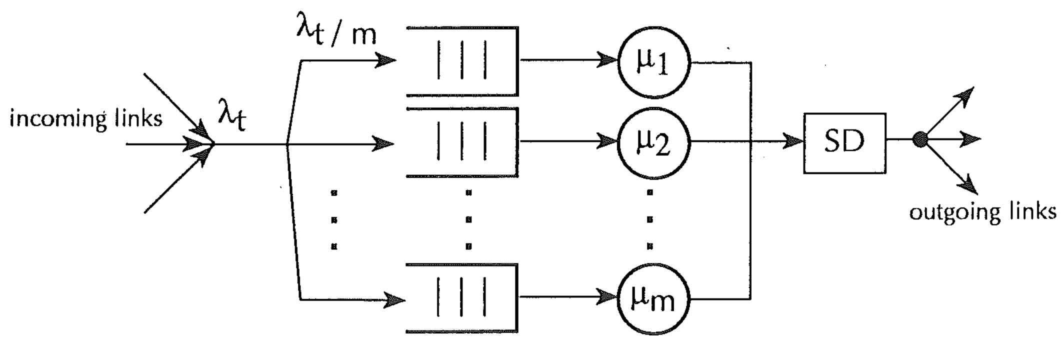

Figure 4.2 Backbone node m-M/D/1 queueing model

The total delay (T_n) imposed by a packet switching node can be determined by:

$$T_n = \frac{2\mu - \lambda}{2\mu(\mu - \lambda)} + SD \qquad (4.2)$$

Where

μ = Mean service rate for a PSC (in PPS).

λ = Mean packet arrival rate for a PSC, i.e. $\frac{\lambda_t}{m}$ from figure 4.2 (in PPS).

SD = Load independent serial processing delay (in sec).

TOPOLOGICAL DESIGN

The determination of a cost-effective topology is one of the most important and difficult design activities facing the network engineer. The process of developing a new topology or the reconfiguration of an existing one represents a tradeoff between the cost and performance quality of the resultant network.

A number of topological design methods have been developed over the last decade. These methods can be divided roughly into two groups, each representing approaches to the two generic topological structures — centralised and distributed.

The choice of topology depends, among other things, on the assumed traffic patterns. If the network is to serve as a data exchange between geographically distributed nodes, then a cost-effective structure can be determined using the methods for distributed network design. If however the network traffic is routed to a single switching centre then centralised structures become more applicable.

CENTRALISED NETWORK DESIGN

Considerable research effort has been spent on the development of centralised network design methodologies, and a vast number of papers are now available [7, 21, 67]. Most are based, in some way or another, on tree structures. One of the most effective[8] can be attributed to Esau and Williams [22].

Essentially the Esau-Williams algorithm searches for the nodes that are the furthest[9] from the central node and connects them to the adjacent nodes that provide the greatest cost saving.

Schwartz [78] and Sharma [81] provide a good coverage of centralised network design algorithms.

DISTRIBUTED NETWORK DESIGN

The topological design of distributed networks is substantially different and far more complicated than centralised structures, and not surprisingly fewer algorithms exist for its solution.

The design of distributed networks was first discussed by Frank et al. in 1970 [28] and since then a number of methods have been developed. In a broad sense, the techniques employed to solve the distributed design problem can be categorised into two main groups namely:

- *Mathematical methods* — A collection of mathematical techniques known as linear programming have been applied to many areas of problem solving including the NDP. Linear programming methods minimise (or maximise) an objective function (i.e. cost) subject to a system of equations (i.e. constraints). The formulation of the NDP as a programming problem necessitates several assumptions that are not very representative of the real world case.

The practical constraints imposed by a real network (e.g. discrete non linear link costs etc.) further increase the combinatorial character of the NDP, making an exact mathematical solution virtually impossible to attain. The only realistic option for addressing the NDP is through the use of heuristic methods.

---

[8] Chandly and Russell [99] report that the Esau-Williams method produced solutions that were within 5% of the optimum.

[9] In an economic sense.

• *Heuristic methods* — Heuristic methods are a class of network design algorithms that try to overcome some of the unrealistic assumptions set by the mathematical methods. Typically, these methods start with a given topology and converge on a solution that is 'fairly close' to the optimum.

Heuristics have been developed to attack the NDP as a series of isolated subproblems — the access problem and the backbone design problem. The access problem consists of the determination of backbone node locations and the generation of the subscriber access network. The backbone design problem addresses the development of low cost backbone configurations.

The subsequent sections discuss the heuristic methods employed by this study in more detail.

NODE PLACEMENT TECHNIQUES

In any network that consists of a large number of geographically dispersed subscribers, the strategy for grouping and accessing the network will play a predominant part in determining the total cost of the network. Not surprisingly researchers have worked extensively on the access problem. Some methods are based on clustering techniques [67] whilst others are predominantly driven by cost.

Two effective heuristics in the latter category are the well known ADD and DROP methods. The ADD algorithm was introduced by Kuehn and Hamburger [59] and represents one of the most successful methods to date.

The ADD method works as follows:

- Tentatively place a node at an unserviced location.
- Generate the subscriber access network.
- Evaluate the cost saving realised by placing the node at that location.
- Repeat this procedure for all unserviced locations.
- Place a node at the location that results in the greatest cost saving.
- Repeat the entire process until no more cost savings can be made.

The DROP method [23] essentially reverses the ADD algorithm, whereby nodes are assumed to be at all possible locations. The procedure then eliminates the nodes that result in the greatest cost reduction.

The ADD algorithm utilizes the same *bottom up* synthesis philosophy as NDW, that is, it starts with a simple network and successively generates more complex structures until the *best* configuration is reached. For this reason the ADD method has been employed by this study.

BACKBONE PERTURBATION TECHNIQUES

Backbone perturbation techniques form an important class of topological design heuristic. These methods attempt to improve an existing backbone structure by sequentially altering (or perturbing) small sections of its topology. If the cost of the modified network is lower and it still satisfies the performance constraints it is accepted and the perturbations are continued on the new structure. Therefore, this class of method search *locally* around a given topology for a lower cost structure that still fulfils the specified performance constraints.

At present there exists no efficient technique for the exact solution of the backbone design problem, although two predominant perturbation methods have been developed; The Branch eXChange (BXC) and Generalised Cut-Saturation (GCS) methods.

The Branch exchange method

The BXC method was one of the first perturbation methods to be developed and is discussed by Frank et al. in [28]. Within this paper they described the application of the BXC method for the optimization of the ARPANET network. The BXC method has also been discussed in [7], [26] and [30].

The essence of the BXC method comprises the following three steps:

- ▼ Select two links, preferably not too far apart.
- ▼ Remove these two links from the network.
- ▼ Add two new links using another combination of the four nodes.

The main shortcoming of this method lies in the fact that it requires the exhaustive exploration of all topological exchanges. In practice this tends to be very time consuming.

The Cut-Saturation method

Gerla et al. [29] have proposed the Cut-Saturation (CS) method for improving backbone designs. The CS method can be considered a sophisticated extension of the BXC method because it selectively considers links, rather than exhaustively trying all exchanges. Backbone links are chosen on their ability to improve network throughput and decrease cost.

The algorithm's goal is to find the least cost backbone topology for a specified delay and throughput constraint. To achieve this, the algorithm attempts to relieve the most congested portion of the topology by adding links in the vicinity of the saturated cut[10], to increase throughput, and by removing links from less utilised areas, to decrease cost.

---

[10] Links are ordered according to their utilization and then successively removed until the network becomes disconnected. The minimal set of these links is termed the *saturated cut.*

The CS method described in [29] was somewhat biased by the ARPANET environment and a more generalised version has since been developed with those earlier deficiencies eliminated. The Generalised Cut-Saturation (GCS) method represents one of the more effective backbone optimization heuristics available today, and has been employed by several researchers [13, 73, 90].

The GCS algorithm has been implemented in NDW and several improvements where made in order to improve its performance. These are discussed in the latter part of Chapter seven.

Although perturbation techniques have been discussed in a vast number of papers — the authors have all failed to highlight the major limitation of these techniques in their ability to determine optimum cost topologies. This limitation is discussed in the next section.

Limitations of perturbation techniques

Perturbation techniques alter the network on a link-by-link basis and only one or two discrete variables[11] change state during any iteration. This alteration process always converges on an equilibrium point which represents a *locally* optimum topology that satisfies the initial design constraints. The important point to note is that this local minimum does not necessarily correspond to the optimum cost topology.

A possible solution to the network optimization problem will be a network n^*. This network configuration (n^*) comprises a large number of variables (e.g. link speed, node positions, backbone structure etc.).

Figure 4.3 details an artificial representation of the solution space for the optimization problem. The initial, intermediate, and final solutions (n^*) and their respective costs $C(n^*)$ all correspond to points on this line.

[11] e.g Link capacity and/or link position.

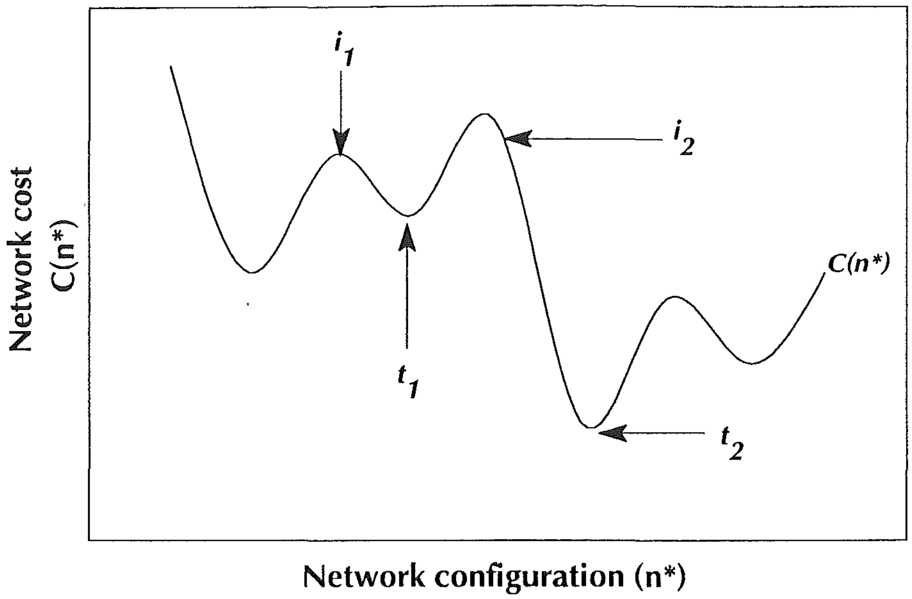

Figure 4.3 Network solution space

Thus, the design process can be viewed as the search for an acceptable configuration (n*) in this very large solution space.

The *distance* of the initial solution (i_1 or i_2) from the optimum configuration greatly influences the chances of whether the global optimum (t_2) or just a local optimum (t_1) is reached.

If the initial network corresponds to a configuration with the optimum number of nodes, i.e. point i_2 in figure 4.3[12], then any reasonable series of perturbations should converge on the optimum solution (t_2). If, however, the initial network corresponds to a more *distant* configuration (i_1), for example one with too many nodes, then the convergence on the optimum structure will depend on the extent of the perturbations. It is possible to make the extent of these perturbations so great that all *values* of n* are covered — this of course equates to the exhaustive search of all topologies.

This study illustrates the need to integrate the node placement and backbone design processes in order to provide a greater coverage of the network solution space (n*), and therefore increase the possibility of the global optimum being reached.

THE NEED FOR INTEGRATED TOPOLOGICAL DESIGN

Conventional approaches to the distributed network design problem have divided it into several subproblems — decoupling the node placement and

---

[12] This point also corresponds to the minima of figures 10.1 and 10.2 in Chapter ten.

subscriber access design from the backbone design process, solving them in isolation, and then combining them for the overall design solution.

The total cost of a communications network consists of the sum of three components; the hardware (switches etc), the backbone network, and the subscriber access network. In reality, the optimum network structure is highly sensitive to the interdependence of these three cost components[13].

One of the reasons for this interdependence is that decreasing the cost of one component will affect the cost of the other two. For example, increasing the number of backbone nodes will decrease the access cost and simultaneously increase the hardware and backbone link costs. This implies that in order to optimize the total cost of the network, the node placement and backbone design components must be treated as a unified optimization problem.

One of NDW's objectives is to minimise the total network cost whilst still satisfying the given design constraints. NDW integrates the access network and backbone design processes with the node placement procedure, in order to synthesise the most cost-effective topology.

A more detailed discussion of NDW's network synthesis methodology can be found in Chapters six and seven.

NETWORK COST ANALYSIS

The accurate determination of network configuration costs plays a predominant part in any design process. The availability of computer-based tools greatly facilitates the speed and accuracy with which these calculations can be made, enabling the designer to rapidly evaluate the economic consequences of changing design variables.

Owing to the numerous economic factors involved in costing a modern network (e.g. non linear multi-vendor line tariffs, modular hardware costs etc.) some form of computerised costing facility is essential for dealing with all but the most trivial of networks.

This section details the network costing methodology that has been developed for this study and implemented by NDW.

NETWORK COST ELEMENTS

The total cost of a communications network can be calculated from the sum of three primary components.

$$C_{net} = C_{ac} + C_{bb} + C_{sw} \tag{4.3}$$

[13] This cost interdependence is illustrated in figures 10.1 and 10.2.

where C_{ac} = Cost of subscriber access network;

C_{bb} = Cost of backbone links;

C_{sw} = Cost of backbone switches.

Each of these three components consists of several sub-components. The calculation of these sub-components are detailed below.

Transmission link cost

The transmission requirements of a link are assumed to be met by commercial leased services (e.g. Telecom, AT&T etc.), so consequently link costs are totally dependent on their pricing structure[14].

In general the cost of a communication link n_i is dependent on four components:

$$C_{link}(n_i) = C_{init} + [\,(C_{access} + C_{trans}) \times P_{rental}\,] \tag{4.4}$$

where

- C_{init} is an initial installation charge levied by the vendor;
- C_{access} is a monthly access or termination charge (usually charged on a per end basis);
- C_{trans} is a monthly transmission charge;
- P_{rental} is the link rental period (in months).

C_{init}, C_{access} and C_{trans} are all dependent on the leased link capacity with C_{trans} being additionally dependent on the link distance.

This costing model can now be used to determine the backbone and access network costs.

Backbone link cost (C_{bb})

The cost of the backbone network can be calculated using equation 4.4 and is simply:

$$C_{bb} = \sum_{f} C_{link}(f) \qquad \text{where } f = \text{set of all backbone links} \tag{4.5}$$

Subscriber access network cost (C_{ac})

The cost of a subscriber access link n_i can be represented by:

$$C_{sub}(n_i) = C_{acclink} + C_{interface} \tag{4.6}$$

---

[14] The transmission link tariffs used by this study are detailed in Appendix I.

where

- $C_{acclink}$ is the cost of the circuit linking the subscriber equipment to the access port on a network switch. $C_{acclink}$ can be calculated using equation 4.4;
- $C_{interface}$ is the cost associated with any terminal interfacing equipment (i.e. PADs, FEP's, modems etc.).

Thus, the total access network cost for n subscribers is:

$$C_{ac} = \sum_{i=1}^{n} C_{sub}(i) \tag{4.7}$$

Backbone switch cost (C_{SW})

The backbone switch (for costing purposes) may be regarded as a termination point for communication circuits, with each circuit requiring an individual port. The more circuits terminating at a node the more ports required and the greater the switching cost.

Modern packet-switching nodes may be expanded by the addition of extra processing modules. Each new module has a fixed cost and supports a fixed number of I/O ports.

Therefore, the cost of a modern switch can be represented by:

$$C_{switch}(n) = C_{base} + C_{modules} + [\, C_{overheads} \times P_{time} \,] \tag{4.8}$$

where

- C_{base} is the fixed base price[15] of the switch hardware (less its processing modules).
- $C_{overheads}$ is the monthly cost associated with housing and maintaining the node. This component is usually variant and dependent on the node's size and location.
- P_{time} is the cost analysis time frame (in months).
- $C_{modules}$ is a variable cost that is dependant on the number of circuits terminating at that node. $C_{modules}$ can be determined by:

$$C_{modules} = \left\lceil \frac{N_{bbports} + N_{acports}}{MAX_{ports}} \right\rceil \times C_{pm} \tag{4.9}$$

[15] Includes the cost of racks, power supplies, cabling etc.

$N_{bbports}$ and $N_{acports}$ are the number of backbone and subscriber access ports terminating at the node. MAX_{ports} is the maximum number of I/O ports a single module can accommodate.

$\lceil x \rceil$ denotes the ceiling function of x — the smallest integer equal to or greater than x.

C_{pm} is the cost of an processing module.

Thus, the total switching cost for an *n* node network is:

$$C_{sw} = \sum_{i=1}^{n} C_{switch}(i) \qquad (4.10)$$

Total network cost (C_{net})

The total network cost is somewhat complicated by the fact that it consists of both acquisition (initial one off expenditures) and monthly lease costs.

The best method of combining both these aspects is to calculate the total cost over a fixed time frame. Besides, any cost analysis is worthwhile only if it is projected over the network life cycle; or at least a good portion of it.

The projection of costs into the future must be augmented to reflect their true costs over that period. This involves several adjustments namely:

❑ *The adjustment of future costs to present value* — To calculate the rental costs of leased equipment over a fixed period of time, one must consider "the cost of money." Equation 4.11 represents an accepted method of determining the present value (PV) of equipment leased over a period of m months [36]:

$$PV = \frac{R}{i}\left[1 - \frac{1}{(1+i)^m}\right] \qquad (4.11)$$

where

R = monthly rental;

i = monthly interest rate;

m = total rental period (in months).

❑ *The projection of future line tariffs*— Several studies [18, 76] have proposed models for the estimation of future line tariffs. These models plot the cost of past link tariffs over a nominal time frame and then use that curve to extrapolate into the future. This trend curve provides a multiplicative factor for the prediction of future tariffs.

This technique is highly dependent on the historical tariff data and only provides a very rough guess at what 'the future holds'. The main problem with these projection models is that there is very little basis available from which to project the costs. They totally disregard the impact of new services, new technology, and the unpredictable influence of competitive factors.

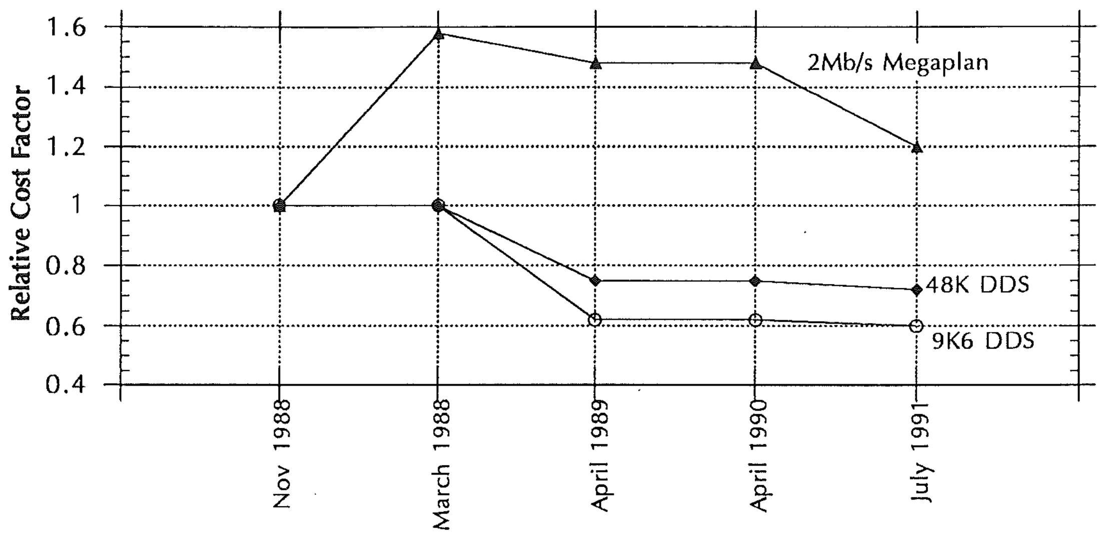

Figure 4.4 Transmission cost trend for Telecom digital links (rental period 6 months)

Figure 4.4 illustrates the cost trend for leasing a digital link from Telecom. From this graph, one can see that the DDS service has decreased by 28-40% over the last three years. The Megalink service however, has increased by approximately 20%!

Unlike overseas markets, New Zealand does not reap the benefits of large competitors to Telecom, thus resulting in a much flatter cost curve compared to those of other studies. Telecom has also just introduced three new services (19K2, 64K and 128K DDS) so at this stage their pricing signals for the future remain unclear. For these reasons this study has used a relative cost factor of 1.0 for the projection of future line tariffs.

CONCLUDING REMARKS

This chapter has highlighted the main techniques used to perform the three primary network design functions — performance prediction, topological design and cost analysis. The successful integration of these methodologies is the goal of designing a Computer-Aided Network Design (CAND) tool.

The next chapter discusses various aspects of CAND. Chapter six introduces Network Designers Workshop (NDW), a design tool that employs most of the methods discussed in this chapter.

chapter five

COMPUTER-AIDED NETWORK DESIGN

The successful planning or re-evaluation of a communication network requires the quick and accurate quantification of many complex technical and economical factors. This mandates the use of computer-aided design tools for all but the most trivial networks.

This chapter provides a look at Computer-Aided Network Design (CAND) and highlights a small set of packages that are representative of the design tools being used today.

The latter portion of this chapter discusses some of the issues surrounding the development of a good network design tool, including several factors that have influenced the design of NDW.

CAND: AN OVERVIEW

BRIEF HISTORY

Communication networks have proliferated over the last two decades. As a result, a large number of engineering tools and techniques have been developed to assist in the analysis and design of these systems. Practitioners were quick to realise the valuable role the computer could play in the design process.

CAND tools fall into two groups — those that are based on analytical models and those that employ simulation methods. This section briefly discusses the history of analytically based CAND tools. Shanmugan [80] provides a similar historical perspective for simulation based tools.

The first packages were based on the theory of queueing networks. These early tools emphasized the generality of the underlying theory by deliberately avoiding bias toward any one physical system. Nodes were described as *service centres*, links as *edges*, and topologies in terms of *branching probabilities*.

Unfortunately it was this generality that increased the difficulty of using these packages. Users were forced to constantly translate familiar terms and concepts into the tool's domain which was both unfamiliar and unnecessarily complex. In this respect these early packages were not successful in their use as network design tools.

The first commercially successful CAND tool was BEST/1 [9]. The developers of BEST/1 recognised the simple fact that its users were network designers and orientated the tool toward that area. The terminology was changed and only the flexibility and generality needed to support network design was incorporated. This reduced the number of input parameters needed for the tool and greatly enhanced its ease of use.

Design tools started to move away from the concept of generality and become much more specialised. They started to incorporate network design libraries that performed specialised functions, such as topological design, cost analysis, and report generation.

The first of these second generation tools, such as NDMS [57] (and even some still existing today!), were designed to operate in a batch mode. The user would specify the design input as a series of text files; the tool would then process this input and write the results to a report file [56]. Unfortunately the network design process does not lend itself to batch processing so the need for user-tool interaction soon became evident. Later tools such as MIND [58] and ACK/TOPS [13] were menu driven (at least in some parts) and provided a limited amount of interactive processing.

Only now is effort being directed toward the development of intelligent, user-friendly design packages that take advantage of the recent advances in computer technology.

DESIGN TOOL SURVEY

A survey and literature search was conducted to establish which tools were already available and how they could be expanded upon. Where possible the tool developers were contacted and demonstration versions of their programs obtained.

Several other reviews and comparisons of popular commercial design tools have been given by Van Norman [94], Kronsynski [57], and Shanmugan [80].

These surveys were supplemented by discussions with the designers and engineers involved with the running of three major New Zealand networks. This unique opportunity enabled the gathering of their informed opinions on the types of tools required to improve the effectiveness of their engineering activities. The key requirement was identified as the need for improved graphical tools that would enable the fast and flexible exploration of network designs.

Name	*Model*	*Reference*	*Date*	*Application*	*Graphics*	*GUI ?*
ACK/TOPS	Both	[13]	1981	Data	Nil	No
Best/1	analytic	[6]	1984	Data	Nil	No
CAPLTN	analytic	[61]	1988	Voice	Nil	No
Credo-HG	analytic	[60]	1988	Data	Partial	No
Criter	analytic	[1]	1988	Voice	Partial	Yes
Data Pathfinder	analytic	[90]	1989	Data	Partial	No
Designet	analytic	[63]	1986	Both	Partial	No
Econets	analytic	[81]	1990	Both	Partial	Yes
HANDS	analytic	[39]	1988	Data	Partial	No
INTER	analytic	[8]	1985	Data	Nil	No
ISCP/NET	analytic	[52]	1985	Data	Nil	No
MIND	simulation	[68]	1989	Data	Adv	No
Minder	analytic	[35]	1988	Voice	Partial	No
Muples	analytic	[88]	1989	Both	Partial	No
NetMod	analytic	[3]	1991	Data	Partial	Yes
NetCon	analytic	[50]	1986	Data	Nil	No
Node/1	analytic	[96]	1989	Voice	Nil	No
NTD	analytic	[38]	1989	Voice	Nil	No
Otarie	analytic	[32]	1988	Voice	Adv	Yes
Topnet	simulation	[64].	1990	Data	Adv	Yes

Table 5.a Design tool landscape

Table 5.a summarises the diverse range of design tools that have been identified from the literature search.

The graphical capabilities of the tools have been assessed and rated as follows:

- *Adv* — Advanced graphics (i.e. the colour coding of the topology and/or performance graphs etc.)
- *Partial* — Partial graphics (i.e. limited to topology display only).
- *Nil* — No graphics; all output is by way of alphanumeric reports.

From this survey, and also the one conducted by Van Norman, it became evident that the majority of design tools were lacking in one or more of the following key areas; Ease of use, graphics capabilities, effective feedback and interaction with data collection software.

It was evident that there was an industry need for new and more productive tools to aid the engineer in the design process. Network Designers Workshop has been developed to address this industry requirement for improved tools.

NETWORK DESIGN TOOLS

The network design tool is essentially a collection of computerised capabilities. It provides facilities for the storage of data (e.g. topologies, hardware profiles, link tariffs etc.), the processing of that data (e.g. flow assignment, delay and utilization calculations etc.), and the presentation of that data (e.g. reports, graphs etc).

This section provides a closer look at a small set of analytical tools that are representative of the current "state of play" in the commercial and research arenas. Shanmugan [80] provides a similar overview for simulation based design tools.

HUGHES AUTOMATED NETWORK DESIGN SYSTEM

Hughes Automated Network Design System (HANDS) [39] was developed by Hughes Network Systems (HNS) for the analysis and design of their X.25 packet switching networks.

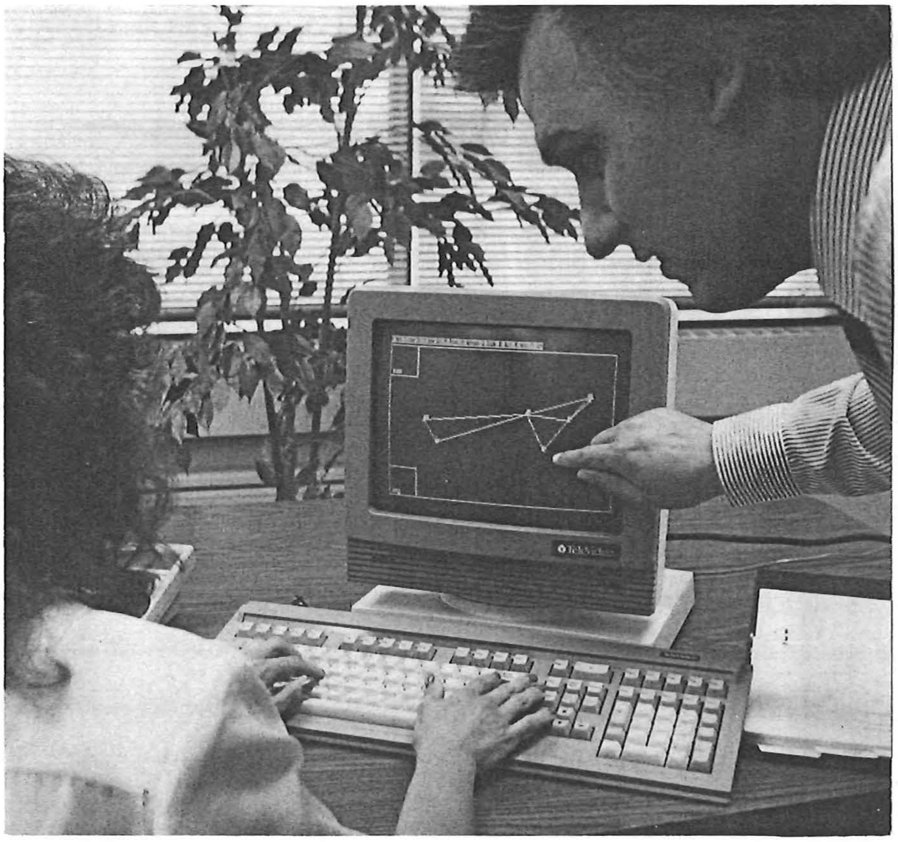

Figure 5.1 HANDS (from [39])

HANDS is a character based menu driven program whereby all user interaction is directed through a series of numbered menu options. Graphics capabilities are limited to a display of the current topology (see figure 5.1).

The tools main disadvantages lie in its lack of user interactivity and design feedback. For example, to evaluate the consequences of adding a node the *"old design is recalled and modules 5 [Design Backbone network], 6 [Price Leased Lines], 7 [Configure Switches], 8 [Compute Performance], and 9 [Generate Report and Graph] are used "* [39].

Additionally, if the designer wants to gauge the economic impact of adding a new host to the network the following process is followed —

- ▼ Module 1 (Input network information) is selected from the main menu and the subscriber file is edited.
- ▼ Module 3 (Assign each site to a backbone site) is used to home the host to a backbone node.
- ▼ Module 7 (Configure switches) is used to reconfigure and cost the access node.
- ▼ Module 6 (Price leased lines) allows the user to cost out the new host access link.

If the user then wishes to compare this cost to the cost of placing the host somewhere else the whole process must be repeated. Obviously this style of operation does not facilitate the rapid exploration of the numerous design options available to the engineer.

DATA PATHFINDER SYSTEM

The Data Pathfinder system [90] is a data network design package designed by Telco Research. Data Pathfinder is written in Fortran-77 and currently runs under DEC's VAX/VMS operating system.

User interaction is through a series of pop-down menus and is based on a 'form filling' metaphor. During any stage of design process the user can retrieve a full screen[1] tabular report detailing the performance characteristics of a particular network entity. Graphics are limited to a topology diagram superimposed onto a map of the United States. The user can choose to zoom in on parts of the topology and generate snapshots of the diagram for an external plotter device.

Data Pathfinder is a powerful and flexible system, however it suffers from the same problem of limited graphics and low levels of user-tool interaction whereby *"Large network design jobs, ... are typically run in batch mode."* [90].

ECONETS

Econets (economical networks) is a software package written for the Macintosh II series of computer [81]. Econets has been developed for the analysis and design of voice and data networks.

On initialisation, Econets loads up to 16 input files to support the design session and presents the user with 5 pull down menus.

According to Econet's author *"it is so user-friendly that the designer can now perform a large number of 'what-if' type topological optimizations in quick succession"* [81]. After a close investigation of this tool, it appears that Econets

---

[1] This implies that several components cannot be compared on the same screen at the same time.

lacks several features that would greatly aid the designer in this process of what-if exploration. For example, to determine the effects of adding a node[2] and access link the designer must perform the following —

- Select VHD from the FUPDATE menu and page to the first blank node description cell.
- Type in the node's number and location (V-coord and H-coord). Once the node data has been entered the *"designer must never forget to add a negative number in the cell next to the last vector[3]"* [81].
- Select LINK from the FUPDATE menu, page to the node's link description cell and enter the link type (an integer between 0 and 4).

Econets presents its results as a series of text files. To determine the performance results of a single network configuration the designer must use the CREATDB menu to consolidate the various support files maintained by Econets. *"Since the output file is saved under a relatively fixed title (e.g., VHD59DB), the designer should rename this output file quickly before it is overwritten by another DB file"* [81].

Graphics capabilities are limited to the display of the current topology in a separate window.

Econets is a flexible tool that can be applied to the design of several network types (e.g voice, data, centralised, distributed, star etc.), however it appears that this flexibility has been provided at the cost of several other desirable attributes such as effective feedback, graphics, and ease of use.

TOOL DESIGN ISSUES

Network engineers soon discover that any network design no matter how sophisticated the tool used will (hardly) ever be perfect. There exists several obstacles that stand in the way of perfection and these include:

- Dependency on the accuracy of the input data;
- Theoretical design heuristics that provide 'less than ideal' solutions;
- The quality of the network design tool;
- The quality of the network designer.

These imperfections ensure that network design remains a combination of human skill and computer brute force. The effective combination or bridging of the two is one of the primary issues surrounding the development of a good design tool.

The following sections highlight some of the key features that play a predominant role in the overall quality of a design tool.

---

[2] The term *node* is used to describe a network subscriber in [81] rather than a backbone switch.

[3] A row of node description cells.

EASE OF USE

The effectiveness of a tool is essentially the ease with which the designer can use the tool to produce the answers they need. The designer should invest minimal effort in learning how to use the tool. Graphical User Interfaces (GUI) are becoming the mainstream interface for software products today. GUIs offer a more intuitive environment than their command orientated counterparts by dispelling the need for the user to learn a command vocabulary. A Graphical User Interface is essential for enabling the designer to interact with the design process.

THE ROLE OF COMPUTER GRAPHICS

The advent of low cost high performance graphics workstations (e.g. SUN, APOLLO, VAXStations, Macintosh Quadra etc.) has meant that many researchers are now focusing their attention toward graphically-orientated design environments.

Several graphically orientated simulation tools have been developed over the last few years [64, 80], however "*there are very few graphically orientated analytic tools*" [3] available today.

Surprisingly the majority of tools are either PC or mainframe based which severely limits their graphical display capabilities [94]. Even the tools that are based on graphics workstations fail to take full advantage of the graphics capabilities offered by their platforms.

Performance results can be displayed, manipulated and examined in graphical form. In addition to the quantitative results, a graphically orientated environment enables the designer to develop a qualitative feel for the network under study. The design becomes a dynamic entity and no longer a static alphanumeric listing on paper.

Colour can also play an important role. Each link and node within the circuit layout can be colour coded according to a variety of attributes (e.g. utilization, delay etc.) thus enabling the user to immediately identify trouble spots in the design. A number of tools utilize this colour coding feature [60, 63] although it is somewhat limited in its ability to display end-to-end data such as response times, average network delays etc.

Fast responsive graphical displays enable the designer to efficiently judge the consequences and quality of the on-going design and note transient effects that would normally be lost in character based environments.

INTERACTIVE ENVIRONMENT

A CAND tool will be used to generate and regenerate cost and performance measures in response to changes in the design parameters. A typical design session will involve a close relationship between the planner and the network design package. A highly responsive environment is central to facilitating this man-machine interaction. Any tool should facilitate the rapid prototyping and evaluation of network designs. Network configurations must be easily altered and the corresponding effects immediately displayed. Finally, the designer should have the ability to quickly store, recall and compare data relevant to many different designs.

CONCLUDING REMARKS

This chapter has provided a brief look at Computer-Aided Network Design (CAND). Several 'state-of-the-art' tools have been investigated and some of their common shortcomings identified.

The following two chapters introduce Network Designers Workshop (NDW), a visually orientated design tool that has been developed to encapsulate some of the desirable features lacking in contemporary tools.

SECTION III

NETWORK DESIGNERS WORKSHOP

Give us the tools, and we will finish the job.
Winston Chirchill 1874-1965
Radio Broadcast, 9 Feb 1941.

NDW: A SYSTEM OVERVIEW

Network Designers Workshop (NDW) is a graphically orientated network design tool which has been developed to:

- Address the shortcomings of contemporary design tools;
- Perform the second phase of the design cycle[1];
- Exploit the graphics capabilities of todays powerful workstations;
- Provide a pragmatic design *workshop* for the network engineer.

This chapter provides an overview of NDW including its design philosophy, underlying architecture, and operational features.

MOTIVATION FOR CONSTRUCTION

There are a variety of network design tools available today which are designed to provide close to optimum network solutions. A review of twenty-six different tools (see Chapter five) revealed that the majority were lacking in one or more of the following key areas:

- Ease of use;
- Graphics capabilities;
- Effective feedback;
- Interaction with data collection software.

The computer-aided planning tool presented in this thesis has been developed to address these shortcomings. NDW enables the designer to specify a number of design parameters and to alter them quickly and effortlessly, with the primary goal of optimizing the network through a process of iterative design (see figure 6.16).

The use of NDW for the re-design of an existing nationwide packet switching network is detailed in Chapter ten.

---

1 Refer to figure 3.1.

DESIGN PHILOSOPHY

Traditionally, large mainframes were used for computer-aided network design, but due to the inherent nature of these platforms, mainframe based design tools lacked most (if not all) of the aforementioned features.

The introduction of powerful graphics workstations over the last decade have provided the network designer with the much needed hardware tools, yet surprisingly few network design programs fully utilize these new hardware capabilities.

The above considerations prompted the development of NDW, a software tool that provides an interactive framework for the analytically-based evaluation and design of packet based communication networks.

The primary design objective was to provide the user with —

- ***A high degree of interaction*** — One of the primary goals was to create an interactive tool which would facilitate the rapid prototyping of network designs. Such a tool should increase the designer's appetite for evaluating alternative network configurations. Any design environment must be sufficiently responsive with the results of any action being immediately displayed. The designer should not be held back while the computer is busy calculating. Fortunately, the advent of powerful workstations and careful coding have made the realisation of this goal possible.

- ***A visual exploritorium*** — The effectiveness of a tool is dictated by the ease with which the designer can use it to yield useful results. Central to facilitating this goal is the provision of an effective bridge between the user and the design tool. Thimbleby [92] points out that *"Human-computer interaction can be improved by: ... increasing the information capacity of the user interface."* The primary means of achieving this is through the provision of a graphically orientated interface.

The use of graphics to display and assimilate statistical data has spread rapidly in recent years. This trend has been highlighted by the popularity of Computer-Aided Design (CAD) packages and the presentation abilities of micro-computer based spreadsheets. Responsive graphical displays enable the designer to efficiently judge the consequences and quality of an on-going design by providing all the information — and then some — which would otherwise have to be extracted from tabular reports. The transformation of results into graphical form will result in impressive gains in the designer's productivity.

Bell et al. [5] note that few rigourous empirical findings on the benefits[2] of visually orientated systems have been reported in the literature. In view of the

---

[2] Do they lead to better decisions?

current trend toward graphical computer interfaces however, this issue is largely academic.

• ***Informative feedback*** — Thimbleby [92] notes also that a "*high [user interface] bandwidth provides lots of information, but is not necessarily informative.*"

During a typical design session the user will alter network variables and observe the consequences. Throughout this process of 'what-if' exploration hundreds of network variables will undergo change. Traditionally, the designer would then be required to review the voluminous output to gain an understanding of the design quality. During NDW's initial developmental stages it was found that the direct monitoring[3] of all network components, in order to capture changes in their state, was impractical.

A mechanism was developed to enable the background monitoring of the design process. NDW has the ability to "look over the designer's shoulder" during the design process by periodically evaluating the current network, highlighting possible shortcomings, and suggesting solutions to them.

NDW uses a rule based approach to automatically verify and evaluate the designer's actions. Rules can be constructed (by way of an editor) to ensure the on-going network design fulfils certain cost, performance, and reliability criteria. If the design strays beyond these constraints, NDW's *design adviser* will notify the user and suggest possible solutions to the problem.

Based on some initial experience with NDW, it appears that the *design adviser* mechanism relieves a substantial portion of the design burden from the network engineer. The user is free to concentrate on the more creative areas of the design.

• ***Links to network management software*** — NDW facilitates the direct importation of operational data from the network management facilities that exist on modern systems. Filters can be developed to extract traffic data from the log files maintained by these systems.

USER INTERFACE

NDW draws on the proven desktop metaphor for its primary means of user interaction. Figure 6.1 shows a 'snapshot' of a NDW desktop during a typical design session.

[3] Using NDW's monitor windows (see pages U-14 to U-17 of appendix III for more details).

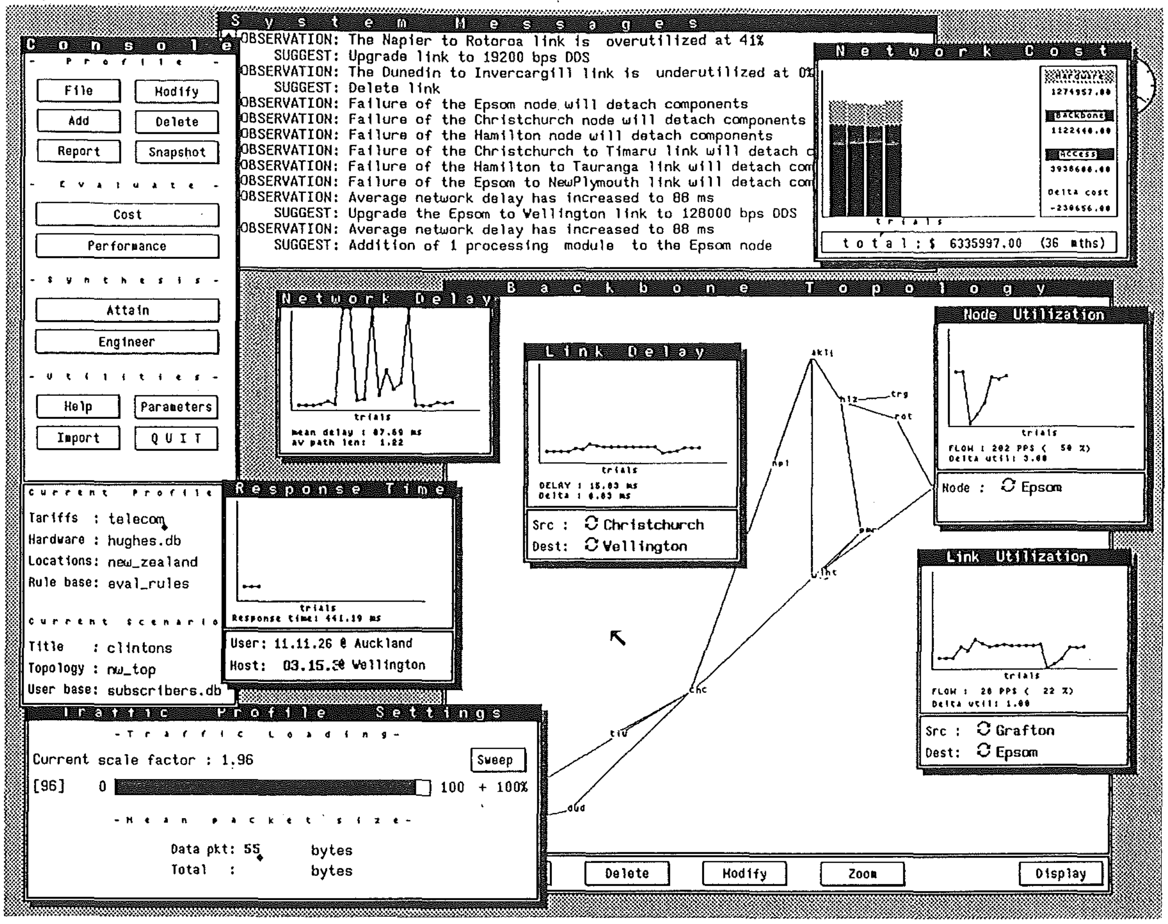

Figure 6.1 The NDW Desktop

Figure 6.1 illustrates the four primary window types maintained by NDW, each of which can be created, moved, iconified and destroyed.

- *Console*— The console window consists of two panels. First, the menu panel where all user commands are issued using the 14 pop-up menus. Second, the status panel which details the names of all currently loaded files.
- *Topology*— The topology window displays the current network topology. Three modification menus offer a range of editing facilities to configure the current topology.
- *System messages*— All system messages and suggestions are relayed to the user through this window.
- *Monitors*— NDW provides several types of monitor windows (several are shown in figure 6.1). These windows enable the user to monitor network attributes (e.g. network cost, node utilization, response time etc.) throughout the design process. The designer can use the monitors to immediately observe the effects of altering network variables. For example, "What happens to the average path length if this link is deleted?" or "What are the consequences of shifting this host, increasing the traffic, altering this node, adding this user?" and so on.

ARCHITECTURAL OVERVIEW

NDW consists of a three layered modular architecture (see figure 6.2).

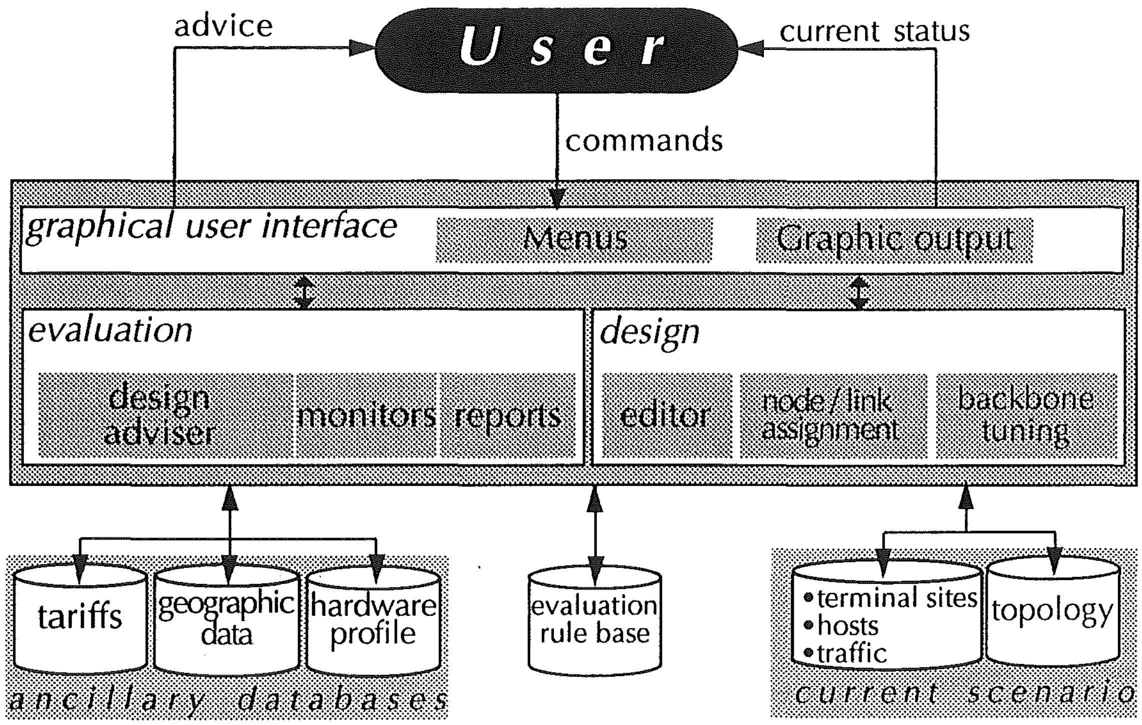

Figure 6.2 NDW System Architecture

Three major architectural layers can be identified in the above figure; the graphical user interface layer, the evaluation and design layer, and the network description layer.

GRAPHICAL USER INTERFACE (GUI) LAYER

One of the most important goals in this project was the provision of a graphical user interface for NDW. Consequently, the first decisive choice was to base the graphical aspects of NDW on the Sun Microsystems graphics library SunView™ (Sun Visual/Integrated Environment for Workstations) [85-87]. SunView is a graphical user-interface tool kit that supports graphics-based applications.

This choice was made for several reasons — First, SunView is a standard windowing environment available on all SUN Workstations. Second, SunView based programs will run under SUN's new Open Windows environment. Third, the SunView libraries are a proven, well documented product that would ensure the rapid and trouble free construction of NDW's interface.

NETWORK EVALUATION AND DESIGN LAYER

The second software layer represents the tool's functional capabilities. It interprets the orders received from the planner through menus, executes the desired task, and passes the results back to the GUI layer in the form of graphs, system messages and reports. This layer can be divided into two main subsystems — network evaluation and network design.

These two subsystems can be subdivided into individual functional groups (e.g. monitors, reports, network editor etc). These functional groups can be further subdivided into individual design functions (e.g. cost node, add link, place node etc). Figure 6.3 contains a complete list of NDW's design functions.

NETWORK DESCRIPTION LAYER

The third layer provides NDW with the design data. Five principle UNIX files are used to maintain the user's design information. Frequently accessed data, such as the traffic matrix, are held in main memory to minimise I/O operations and therefore increase operational speed. The data files contained within this third layer can be divided into the following three categories:

Ancillary databases

• *Tariffs* — contains a detailed description of the leased link vendor charges. The tariff file is used for the pricing of both backbone and subscriber access links[4].

• *Geographic location data* — maintains a list of geographical site names along with their latitude and longitudinal coordinates.

• *Hardware profiles* — maintains an inventory of the available network hardware and their corresponding attributes (e.g. cost, processing delay, capacity etc).

Evaluation rules

This file contains the rules used by NDW's design adviser system. The separation of these rules from the tool provides the user with the ability to edit and change NDW's evaluation criteria without resorting to programming or recompiling.

Current scenario files

A network scenario represents the fundamental network design unit. NDW enables the user to build up and store numerous network scenarios, each consisting of the following files:

• *Subscriber profile* — This file contains a list of all the subscribers currently attached to the network, along with their location, access link type, name, traffic submitted, traffic destination etc.

---

[4] Appendix I details the link pricing structure used in this study.

• *Current topology* — This file maintains a description of the current network configuration including the backbone topology, traffic profiles, node types etc.

MODULAR DESIGN

NDW was constructed in a modular fashion, not only for ease of development, but also to ensure enhancements to the tool could be continued indefinitely.

The modularity of NDW exhibits itself in two distinct ways:

• *The program code modules and their interaction* — Modularization facilitated the process of NDW's development[5] and its continued enhancement. Much attention can be spent on the development of a function, while its integration into NDW can be achieved with little difficulty.

• *The structured user interface* — The practitioner can use a small section (or module) of NDW to effect a particular design goal (e.g. cost a backbone link) or alternatively, integrate a combination of modules (e.g. link and node placement, backbone tuning, monitors etc.) for a complete network design.

FUNCTIONAL OVERVIEW

The following paragraphs provide an overview of the design facilities provided by NDW. For an exhaustive treatment, the reader is referred to the second section of Appendix III. NDW provides the user with 82 design functions logically contained within 19 popup menus. NDW's command hierarchy is represented by figure 6.3.

[5] At the time of writing NDW consisted of 15,671 lines of C code.

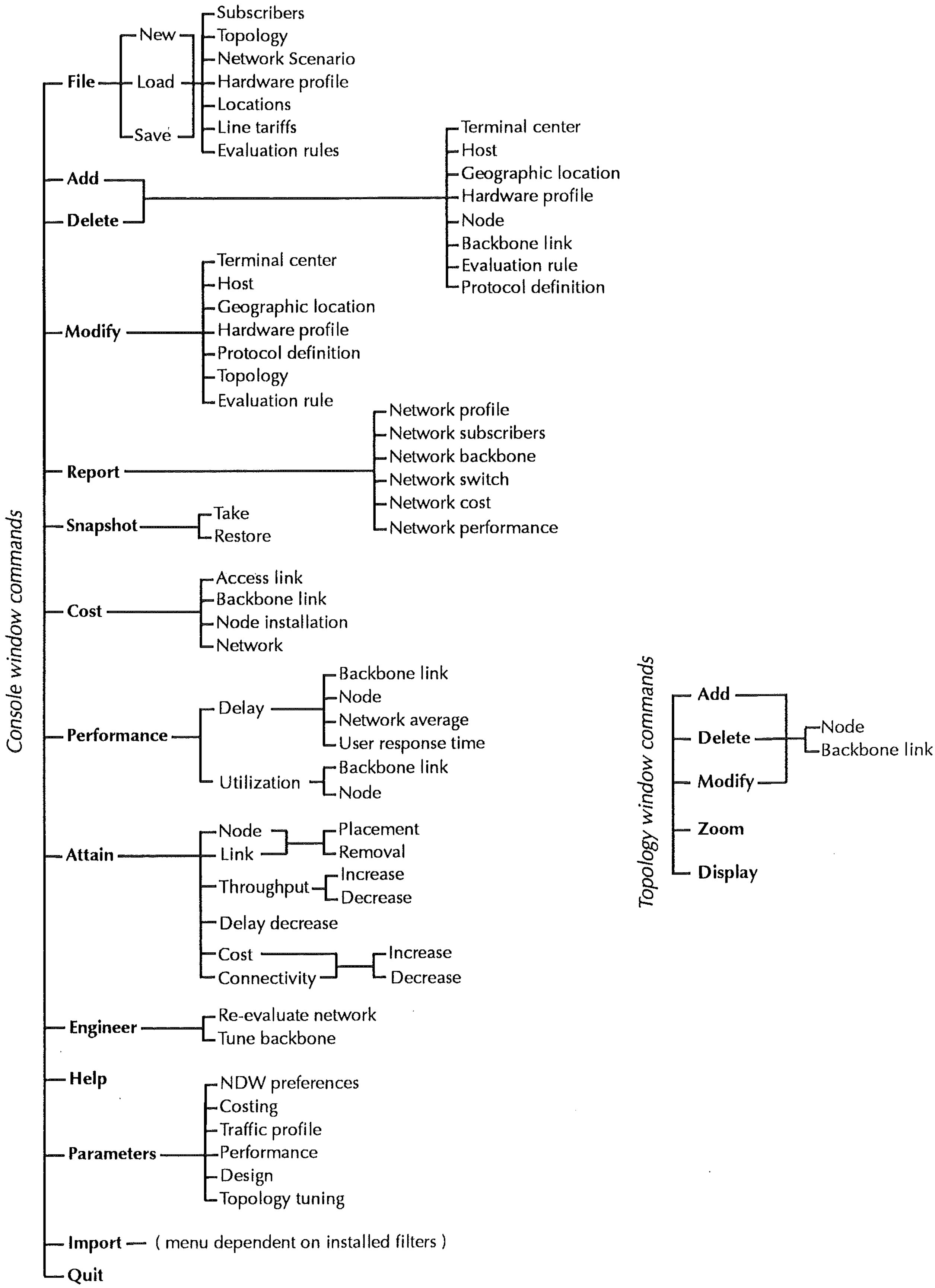

Figure 6.3 NDW command hierarchy

NDW's design functions can be broken down into three main sections: facilities for the description of network profiles, facilities for the evaluation of existing networks, and facilities for the synthesis of new network structures.

NETWORK DESCRIPTION

The setting up of a network profile is the first step in any design process. NDW stores network information as a series of files. Several different groups of these files can then be saved and reloaded to form design scenarios. The designer can then use these scenarios to keep track of different design alternatives.

The network description is built up using the *add*, *delete* or *modify* menu commands. The consequences of any network alteration are immediately displayed within the currently active monitor windows. If the user has a rule-base loaded, the design adviser will evaluate the network accordingly.

A network description consists of the following primary components: network scenario, user profile, topology and hardware descriptions. The following sections discuss each of these in turn.

Network scenario definition

The pop-up *file* menu provides the user with the ability to create, load, and save scenario description files.

Network user profile definition

NDW maintains information on every subscriber connected to the current network. The user can, at any time, add, delete, or modify subscribers by way of the dialog boxes shown in figures 6.4 and 6.5

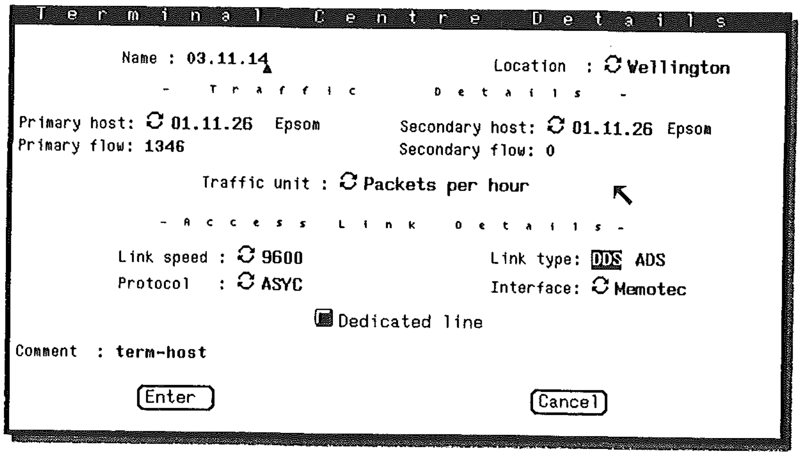

Figure 6.4 Terminal centre definition dialog

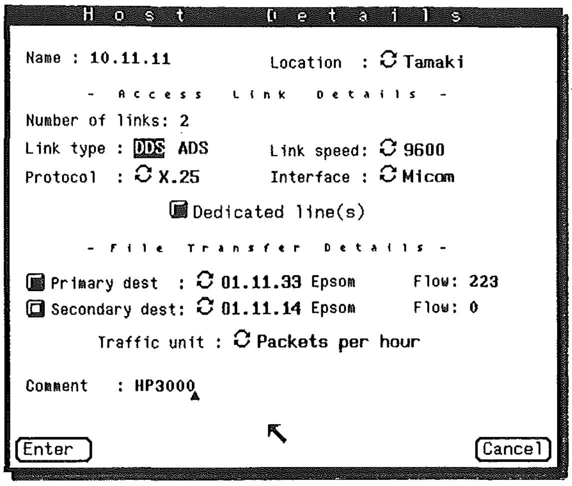

Figure 6.5 Host definition dialog

A full description of the dialog box fields appears on pages U-9 and U-10 of Appendix III.

Network topology definition

The current network topology is detailed in the dedicated windows shown in figures 6.6 and 6.7. The user can define or modify the topology using the *add*, *delete* or *modify* menus that appear on either the main console or the topology window.

After some initial experience with the tool, it was found that the ability to display the topology both geographically (figure 6.6) and logically (figure 6.7) was essential. This ability to strip away the geography from the topology proved useful in complex backbone designs that involved intracity meshes.

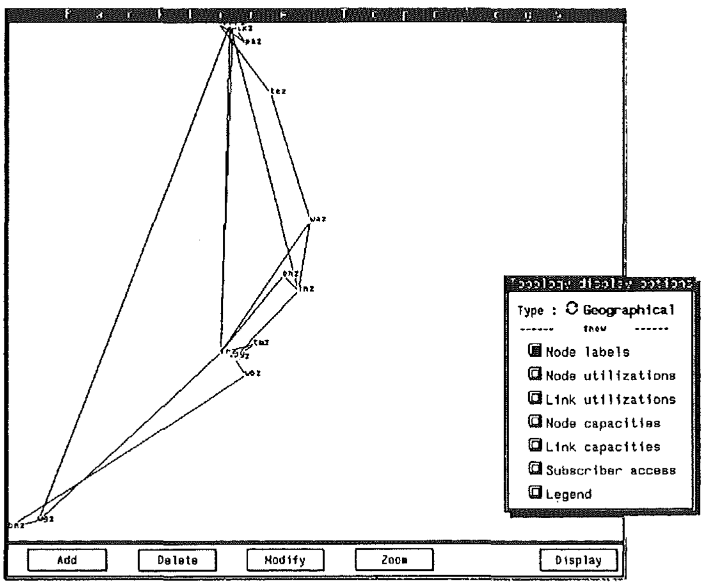

Figure 6.6 Geographical display

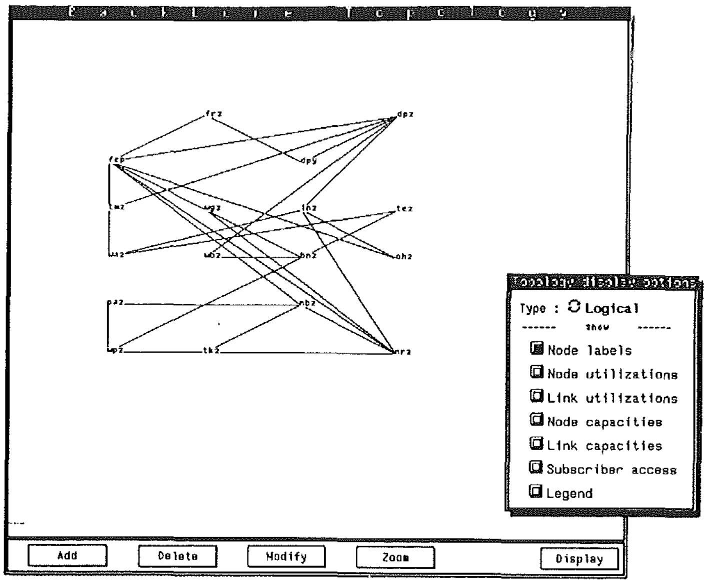

Figure 6.7 Logical display

Network hardware definition

NDW maintains files that contain profiles of network hardware objects (e.g. nodes, PADs, FEPs etc.) and their corresponding attributes (e.g. cost, maximum throughput, maximum number of ports etc.). By entering these profiles, NDW effectively builds an inventory of the equipment that can be used in a specific design. Hardware profiles can be defined using the *add*, *delete* or *modify* menus and the subsequent dialog box shown in figure 6.8.

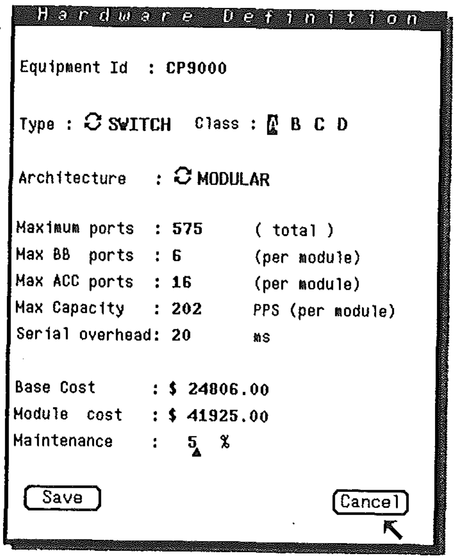

Figure 6.8 Hardware definition dialog

A description of each dialog box field can be found on page U-11 of Appendix III.

NETWORK EVALUATION

NDW is primarily a tool for evaluating existing computer networks. During a typical design session the user will investigate hypothetical situations by altering network variables (e.g. shift a host, increase the traffic, delete a link etc.) and observe the effect of doing so. NDW has two primary mechanisms that facilitate this process of 'what-if' exploration.

Monitors

One of NDW's key features is its ability to selectively monitor network components during the design stage. The *cost* and *performance* menus allow the designer to monitor the following attributes — total network cost, access and backbone link costs, node costs, user response times, link/nodal delays and finally, link/nodal utilizations.

The user can reconfigure the network and immediately observe the effect using one of the eight monitor types provided by NDW. Unfortunately it is difficult to get a feeling for the interactivity of NDW without watching the system being used— no amount of pictures will overcome this problem.

Figures 6.9 to 6.12 detail four of the eight monitors provided by NDW. A complete description of these monitors can be found in Appendix III.

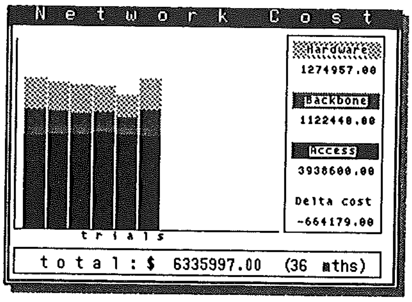

Figure 6.9 Network cost monitor

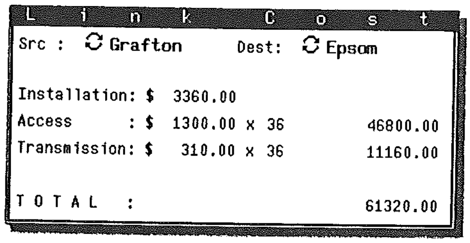

Figure 6.10 Link cost monitor

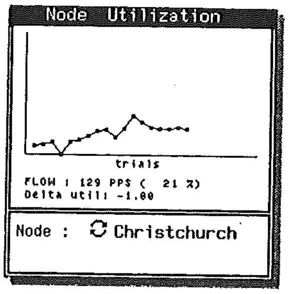

Figure 6.11 Node utilisation monitor

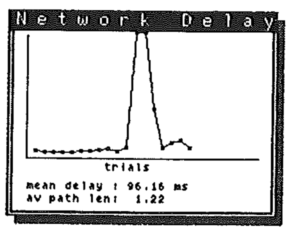

Figure 6.12 Network delay monitor

The monitor graphs are designed primarily for showing *trends* over a series of design iterations, and thus *"provide a very rich form of visual information"* [72].

The monitor windows interleave numerical results with pictorial graphics to provide the user with both qualitative and quantitative feedback.

Design adviser

During a design session hundreds of variables constantly undergo change, resulting in an explosion of information. This makes it very difficult for the designer to maintain a clear understanding of the network's status at any given time. The challenge was to develop a mechanism which would derive the same conclusions about the on-going design as an experienced engineer who had scanned the numerous pages of design reports.

NDW draws on Artificial Intelligence (AI) and Expert System (ES) techniques to provide a powerful yet flexible way of relieving the user from some of the design burden. NDW's *design adviser* uses a rule based approach [62] to automatically verify and evaluate the quality of the on-going design.

A schematic of NDW's design adviser is detailed in figure 6.13.

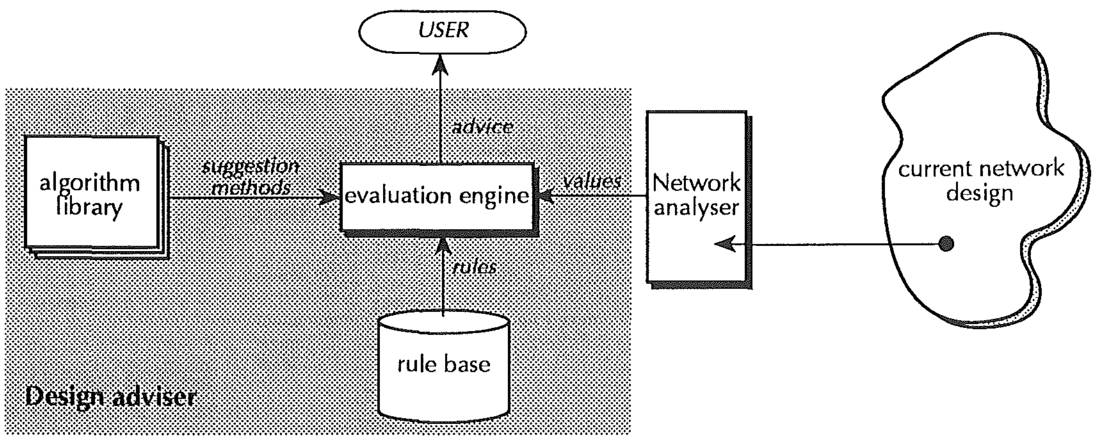

Figure 6.13 NDW design adviser schema.

The design adviser consists of three primary components: The evaluation rules, the evaluation engine and the algorithm library.

• *Evaluation rules* — The rule base contains evaluation rules which combine values obtained from the network analysis component with suggestion methods from the algorithm library. Rules consist of two components; antecedent and consequent. The rule's antecedent component defines its applicability to a particular design situation and the rule's consequent component defines the method for determining a solution to the design situation.

Evaluation rules can be constructed from a set of primitives (using the editor shown in figure 6.14) to ensure the on-going design fulfils certain criteria. Should the design stray outside the constraints set by those rules, NDW will notify the user via the system messages window shown in figure 6.15 and suggest the 'best' solution.

Rules are constructed from a predefined set of primitives and are of the form:

IF: <object> <attribute> <value> THEN <consequent>

At this stage seventy-four different rules can be constructed to evaluate the network and provide suggestions for alleviating the following problems:

- Excessive costs, delays, utilizations and connectivity;
- Insufficient throughput, delays and utilizations;
- Low node/link connectivity.

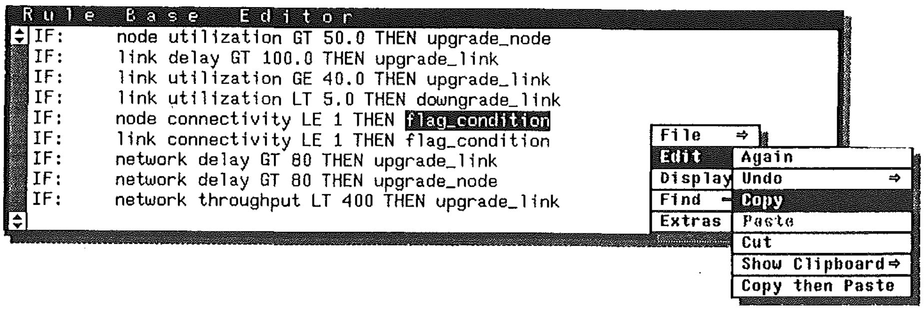

Figure 6.14 Rule base editor

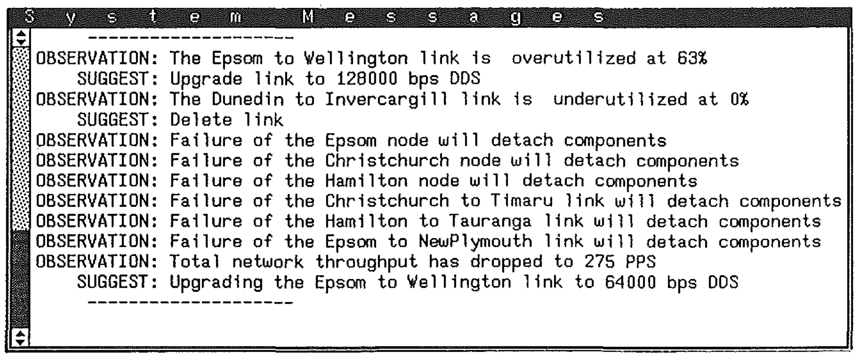

Figure 6.15 System messages window

NDW's evaluation rules are constructed from a predefined group of primitives and must conform to the syntax detailed overleaf.

<rule> ::= (IF <antecedent> THEN <consequent> [ELSE <consequent >])
<antecedent> ::= <object> <attribute> <value>
<value> ::= <comparator><number>
<comparator> := LE I EQ I GT I GE I NE I LT
<object>::= node I link I network I subscriber
<attribute>::= utilization I throughput I delay I connectivity I cost
<consequent> := upgrade_node I upgrade_link I downgrade_node I downgrade_link I upgrade_component I downgrade_component I flag_condition

The *<consequent>* primitives draw on the methods defined in the algorithm library and are dependent on the rule's context; that is the *add_link* primitive uses a different algorithm when used as a consequent in a connectivity rule to those used in a utilization rule. The following paragraphs illustrate several examples of rule construction.

If the designer wants NDW to monitor network throughput, and to suggest the most cost effective link to upgrade (or even add) should it fall below 200 Kb/s, then the following rule would be added to the rule base:

IF: network throughput LT 200 THEN upgrade_link

Alternatively, should the user wish to know if any internodal delays exceed 50ms, without NDW suggesting a solution, the following rule is added:

IF: node delay GT 50 THEN flag_condition

If the user then decides that a 45ms delay should be considered excessive, the rule can be edited thus:

IF: node delay GT 45 THEN flag_condition

If the user does not want NDW to monitor nodal delays, the rule can be removed from the rule base. Thus, specific rule bases can be built up to tailor an individual design effort.

- *Evaluation engine* — The evaluation engine controls the design adviser system. It executes the following steps after each action is made by the user.
  - ▾ Select the evaluation rules pertinent[6] to the last action.
  - ▾ Analyse the network to generate the necessary information for those rules.
  - ▾ Test the analysis values against the rule's antecedent component. If necessary trigger the algorithm specified by the rule's consequent component.
  - ▾ Display the suggestions provided by the consequent component.

---

[6] Rules are selectively chosen, i.e. connectivity rules are not used if the last action was to increase the traffic level.

• *Algorithm library* — NDW's algorithm library contains the routines for determining solutions to design shortcomings. These routines are based on published methods and at present provide solutions for the following three areas: topology [2, 48, 83, 89], performance [7, 17], and cost [78].

The design adviser's algorithm library is constructed in a modular fashion so new methods can be integrated with the minimum of effort resulting in an ever expanding suite of solution methodologies.

NDW's design adviser is designed as a decision support system. Its primary function is to highlight problem areas and suggest options for the user to follow. This system can be thought of as *symptom based*, since the evaluation rules consist of mapping design shortcomings to final suggestions.

Reports

At any stage during the design process the user can use NDW's reporting module to generate comprehensive reports. The user selects the specific report from the *reports* menu — NDW then writes it to a standard text file in a format that is both human readable and directly importable by microcomputer based graphing packages.

The command reference section in Appendix III contains detailed descriptions of these reports.

NETWORK SYNTHESIS

In addition to evaluating an already existing network, NDW can be used to synthesize a new network structure. In such a case, the user would firstly define the subscriber profile using the host and terminal centre specification procedures. The facilities detailed in the following section would then be used to design a cost-effective network structure. The network's specific cost, performance and reliability attributes may then be analysed using NDW's evaluation facilities.

A detailed look at NDW's network synthesis mechanisms can be found in Chapter seven and Appendix V.

The NDW design process

The NDW design and optimization process considers all the cost components stated in equation 4.3 of Chapter four in an integrated manner. Although this is not the first time the backbone design process has been done in conjunction with the node placement process [15], much of the existing literature still treats them separately.

The objective of the NDW design process is to minimize the total cost of the backbone, switching, and subscriber access costs whilst still maintaining the required design constraints. In order to achieve this a series of cost versus network dispersion studies are made until the optimum tradeoff point is reached.

Figure 6.16 illustrates a schematic of the NDW methodology for synthesizing cost effective networks.

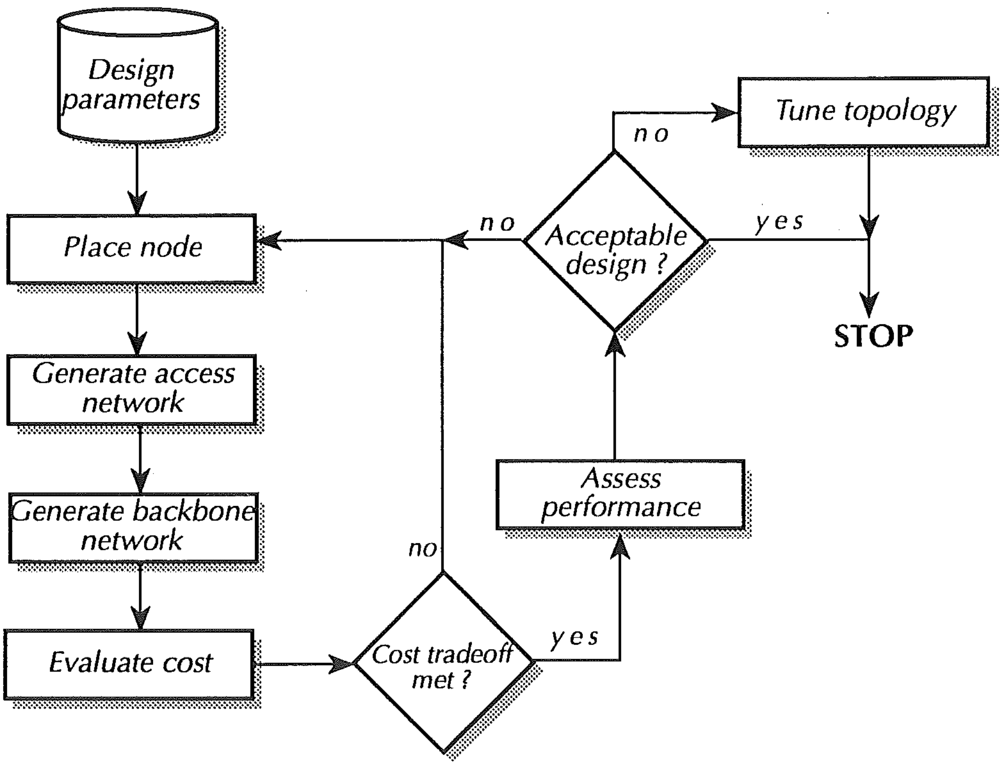

Figure 6.16 NDW design process

The NDW synthesis process, once given a set of design parameters (e.g. maximum link/node utilisations, delays, throughput etc.), employs the following steps:

▼ *Place node* — Determine the best location for the next node (initially the first node).

▼ *Generate access network* — Given the new node set, home all subscribers to the nearest nodes[7].

▼ *Generate backbone network* — Generate an initial backbone structure given the latest node set using the method detailed in Chapter seven.

---

[7] This procedure generates a MCMS topology. MSMD topologies could be generated by employing multi-drop design heuristics to home the subscribers to the backbone nodes.

- *Evaluate cost* — Determine the total cost of the current network configuration using the methods detailed in the latter part of Chapter four.
- *Cost tradeoff met ?* — Has the optimum tradeoff between node, access and backbone costs been reached? If not then place another node and continue the process.
- *Assess performance* — Evaluate the performance of the current network using the methods detailed in Chapter four.
- *Acceptable design ?* — This is a decision that can be made only by the designer (with a little help from the tool). An acceptable design could correspond to the optimum cost tradeoff point (i.e. the minima in figures 10.1 and 10.2) or some other point on this graph that represents a more expensive but more desirable[8] network configuration. If the design is not acceptable then either a new node is placed or topology tuning techniques are employed.
- *Tune topology* — Topology tuning refers to the adjustment of the initial topology suggested by NDW in order to lower cost, overcome congestion caused by shifts in traffic routing, and increase reliability. The designer can use the backbone tuning methods built into NDW to perturb the current topology and find a local cost minimum, or combine their skills with NDW's monitoring and design adviser facilities to engage in a series of what-if explorations.

The following sections discuss the three primary synthesis facilities provided by NDW: node placement, backbone generation, and topology tuning.

Node placement

NDW's *Attain node placement* command can be used to determine[9] the 'best' location for installing a new node. NDW will list all possible node locations and their corresponding cost savings.

The designer can either follow NDW's recommendation, or decide to place a node in some other location. This facility can be used to add a new node to an already existing topology, or to determine the optimum node locations for a new network.

Generate backbone

Given the current nodal locations and the traffic requirements between them, the *Attain link placement* command can be used to find possible ways of interconnecting them.

---

[8] In terms of reliability, expandability etc.

[9] Using the ADD algorithm detailed in Chapters four and seven.

NDW will list possible links[10], along with their traffic requirement, cost, and a preference factor. This factor ranges from one (highly recommended) to zero.

Topology tuning

Topology tuning refers to the process of altering the initial backbone structure suggested by NDW to improve its cost-performance characteristics. Normally this process would be performed only after successive iterations of the NDW synthesis method (figure 6.16) have been applied and the optimum number of nodes determined. By this stage the network configuration should correspond to a point[11] that will, with further tuning, converge on the global optimum.

The user can instruct NDW by way of the *Engineer* menu to employ perturbation techniques to further improve the initial backbone structures suggested by NDW. NDW uses a modified[12] version of the generalized cut-saturation heuristic [17] for tuning backbone structures.

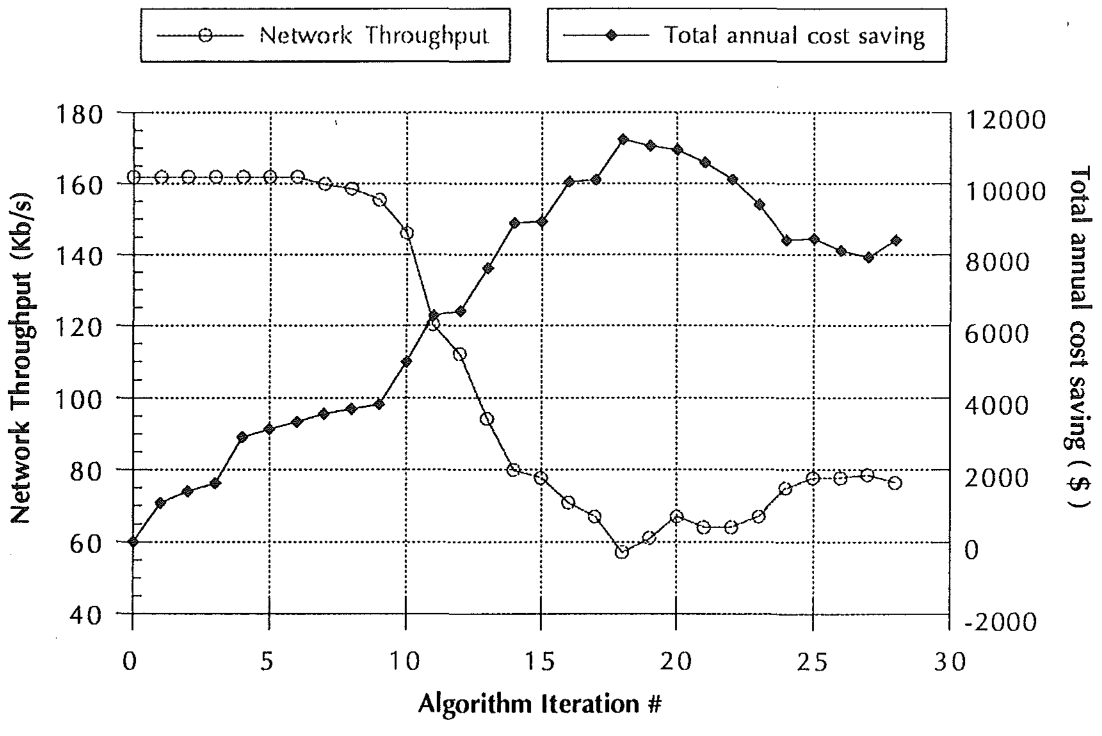

Figure 6.17 NET-X topology tuning trace

[10] Using the backbone generation method detailed in the next chapter.

[11] i.e. point i_2 within figure 4.3.

[12] The modifications made to the generalized cut-saturation algorithm are discussed in Chapter seven.

Figure 6.17 details a trace of the generalised cut-saturation algorithm tuning the original NET-X topology given the following design constraints:

- maximum throughput = 140Kb/s
- minimum network throughput = 60Kb/s
- maximum average network delay = 80ms

One can conclude from the above trace that the use of backbone perturbation techniques can offer improvements on existing topologies (in this case saving $8200 annually), yet surprisingly they do not result in major shifts in the backbone connection pattern. Chapter ten will illustrate how the integrated design technique adopted by NDW results in significant changes in the topology and offers far greater cost savings than the sole use of perturbation techniques.

CONCLUDING REMARKS

This chapter has presented Network Designers Workshop, a graphical tool for the design and performance evaluation of wide area distributed networks.

The following chapter will provide a more in-depth look at some of NDW's implementation details, including the verification of the NDW model.

Chapter ten will focus on the use of NDW for the synthesis of a superior NET-X topology.

c h a p t e r   s e v e n

NDW: IMPLEMENTATION ISSUES

The design and development of NDW involved the use of several classical design methods. This chapter provides a detailed look at the underlying synthesis methods employed by NDW. Several inherent shortcomings of Chou's generalized cut-saturation heuristic are revealed and modifications to it proposed. Finally, several issues surrounding the design of NDW, including the verification of the NDW model are discussed.

OPERATING REQUIREMENTS

NDW was developed using the C programming language [51] and is designed to operate on a SUN™ Sparcstation running the UNIX™ operating system. NDW requires the SunView™ or Open Windows™ windowing environment. In order to fully utilize NDW's graphics capabilities a colour monitor is required.

A typical design session will involve many thousands of floating point operations, making a floating point accelerator chip mandatory for acceptable responsiveness. A minimum of 1.25 megabytes of virtual RAM and 500 kilobytes of disk space are required to run NDW.

Disk space is required for each design project maintained by NDW and is dependent on the size of the network under study.

NDW SYNTHESIS METHODOLOGIES

NDW draws on published methods for its underlying functionalism, most of which have been outlined in Chapter four. The following sections provide a more detailed look at the methods adopted by NDW to execute the design process detailed in figure 6.16.

FLOW ASSIGNMENT

The flow assignment procedure defines the method used to determine the path over which traffic should flow from one site to another. Two types of flow assignment exist — those actually implemented by the network and those used during the design phase. Although many flow assignment methods have been proposed for the latter purpose, a good method must be a compromise between the following requirements [16]:

- It must make use of the available link capacities. In other words, the average delay experienced by the packet must be minimised.
- The assignment procedure is applied hundreds of times during a design session and must therefore be computationally efficient. This requirement effectively rules out discreet event simulations.
- The routing procedure must be similar to the one employed by the final operating network.

NDW's flow assignment mechanism is based on *Floyd's* algorithm [25] for determining shortest paths. This method is very fast to implement and gives a result within 5-20% of the optimum [16]. Additionally, and most importantly, this mechanism closely mimics the actual routing procedure employed by the NET-X network.

NODE PLACEMENT

NDW's node placement mechanism is based on a modified[1] version of the ADD heuristic and consists of the following seven steps:

① Let *sites(i)* = the i^{th} possible node site. Where *sites* represents the set of n possible node locations.

② Choose an unserviced location *sites(i)*, where $1 \leq i \geq n$.

③ Tentatively place a node at *sites(i)* and reassign[2] the subscriber access network.

④ Resize all existing nodes in accordance with their new traffic carrying requirements, that is:

Let N_p = the number of link ports needed for the access and backbone networks.

Let C_n = the traffic carrying requirements of the node.

Size the node with the smallest number of processing modules needed to service N_p and C_n and yet still fulfil the utilization constraints.

⑤ Calculate the change in total network cost realised by the placement of the node at *sites(i)*.

⑥ Repeat steps ② to ⑤ until all members of *sites* have been evaluated.

⑦ Suggest placing a node at the location *sites(i)* that results in the greatest cost saving.

---

[1] The ADD algorithm assumes fixed cost nodes. In reality it is not unusual to have multiple node types (with varying characteristics, capacities, and costs) included in the design. Thus step 4 has been added to accommodate variable node types and sizes.

[2] In this study subscribers were homed to the nearest node resulting in MCMS topologies, although MCMD topologies could be generated if multi-drop techniques are used to reassign the access network.

BACKBONE GENERATION

Each successive iteration of NDW's synthesis method (see figure 6.16) requires the regeneration of the backbone to accommodate the new node and the subsequent shifts in traffic requirements. Although there are no exact solutions for the generation of backbone topologies, one reasonable approach can be attributed to Chou [15]. This method is based on the observation that good backbone configurations are dependent on traffic requirements and link cost structures. A modified[3] version of Chou's method has been implemented by NDW and is detailed below.

❶ Let $trm(k)$ = k^{th} largest element in {TR(i,j)}. Where TR(i,j) is the traffic requirement from node i to node j.

❷ Let TRM = {$trm(k)$ | k =1,2 ... N} be the set of N node pairs with the largest traffic requirements where N maximizes:

$$\frac{trm(N^*)}{trm(N)} \leq A$$

initially $N^* = 1$. A can be set to ∞ for small networks.

❸ Let TRCST = {$cst(k)$ | k = 1,2 ... N} where $cst(k)$ is the cost of servicing the requirements between nodes i and j (represented by TRM(k)) with a link of capacity C_k.

Where $\dfrac{trm(k)}{C_k} \leq U_k$, U_k is a utilization constraint set by the designer.

Let $trc(k)$ be the k^{th} smallest element in TRCST.

❹ For n = 1,2 ... N consider link k whose cost is represented by $trc(k)$

Add the link with capacity C_k to the topology .

❺ Do ❹ until the network is connected.

❻ If n = N and the network is not connected set N^* = N+1 and go to ❷

The designer can now add the links suggested by this process and advance to the next phase in the NDW design process.

TOPOLOGY TUNING

The generalized cut-saturation method, previously discussed in chapter four, has been implemented in NDW to tune final topologies. The algorithm is quite complicated (requiring several hundred lines of code to implement), so a simplified description is presented below. For a detailed discussion see [17].

[3] Chou's method assumes that least cost and minimum length links are homogeneous, and therefore bases step 3 on link distance. This assumption is not realistic and so step 3 has been altered to incorporate link cost.

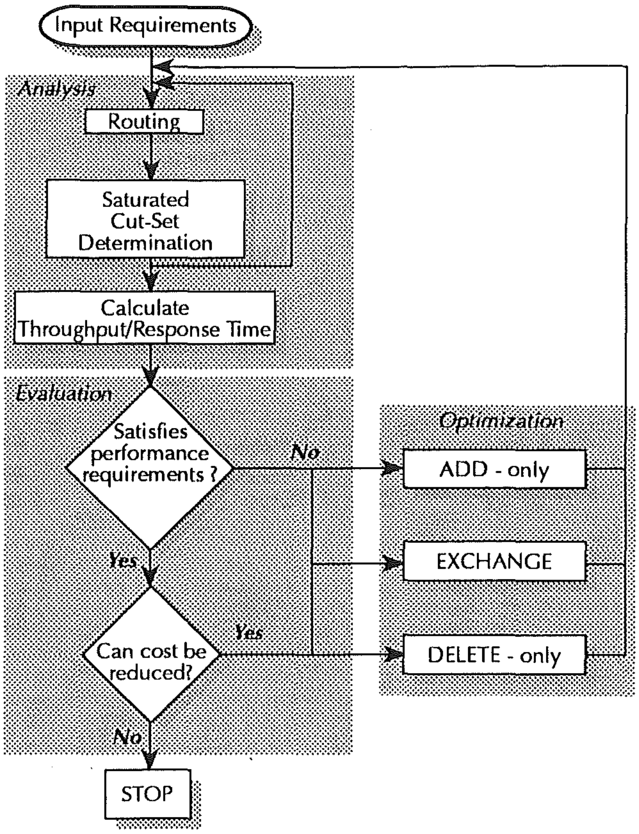

Figure 7.1 Generalized cut-saturation algorithm flow

The generalized cut-saturation algorithm comprises six primary steps in any one iteration namely:

- *ROUTING* — The flow assignment procedure is applied to assign traffic loads on all network components.
- *SATURATED CUTSET DETERMINATION* — Links are ordered according to their utilization. The minimal set of these highly utilized links that disconnects the network is the *saturated cutset*.
- *NETWORK EVALUATION* — The network is evaluated on the basis of upper and lower throughput bounds. Structures with throughputs less than the lower bound are deemed unfeasible and the ADD-only procedure is applied. Networks with throughputs greater than the upper bound are deemed too expensive and the DELETE-only procedure applied. The EXCHANGE procedure is applied to networks with throughputs within the stated bounds.
- *ADD-only OPERATION* — This procedure determines the *best* link to add[4] across the two components separated by the saturated cutset. The link that has the highest ratio of added capacity to added cost is chosen to add across the saturated cut, that is upgrade the link which minimizes equation 7.1.

[4] The 'adding' of a link is used in a generic sense to mean the increasing of the link's capacity to the next available speed. This includes the addition of links which did not exist before.

$$\frac{\textit{new link capacity - old link capacity}}{\textit{new link cost - old link cost}} \qquad (7.1)$$

▼ *DELETE-only OPERATION*— This procedure is applied to reduce the cost of the topology. The link that is not in the saturated cutset and results in the highest cost saving per bit is downgraded[5], that is the link which maximizes equation 7.2.

$$\frac{\textit{old cost - new cost}}{\textit{current link flow - new capacity}} \qquad (7.2)$$

▼ *EXCHANGE OPERATION*— This procedure seeks to improve network cost while keeping the throughput between the specified bounds by combining the ADD-only and DELETE-only procedures. One link is upgraded according to equation 7.1 and one link downgraded according to equation 7.2.

Figure 6.17 in Chapter six details a trace through eighteen iterations of the algorithm. Iterations 0-18, 19-20 and 21-28 correspond to the DELETE-only, ADD-only and EXCHANGE steps respectively.

MODIFICATIONS TO THE CUT-SATURATION ALGORITHM

After some initial experience with the generalized cut-saturation algorithm, several shortcomings were revealed. The main problems lie in its link choice metrics (equations 7.1 and 7.2). It was discovered that the metrics proved to be less than ideal when paired with 'real world' pricing tariffs.

ADD-only metric limitations

Chou's original ADD-only metric (equation 7.1) resulted in several[6] potential links ending up with the same value. There needed to be an additional criterion added to the metric to narrow this 'best choice' down to one link.

DELETE-only metric limitations

The original DELETE-only metric (equation 7.2) suffered from the following limitations:

- What happens when the link flow and new capacity are equal to zero?
- A link will never be downgraded to 0 (i.e. deleted) if there is traffic flowing, no matter how minuscule.

---

[5] A link that is degraded to a capacity of zero is deleted.

[6] Up to seven in the NET-X case.

- Once the algorithm entered the EXCHANGE phase it tended to concentrate on the expensive links and totally ignore any low cost links with small traffic flows.

The following metrics were developed during this study to alleviate the aforementioned problems. The new metrics now provide the designer with greater flexibility through the inclusion of a connectivity measure.

New ADD-only metric

The *best* link to add or upgrade is based on its Link Upgrade Index (LUI):

$$LUI = W_1\,(1-\hat{M}_1) + W_2\hat{M}_2$$

where

W_n is the weight assigned to $\hat{M}_n$ $\qquad 0 \le W_n \le 1$

$$M_1 = \frac{\textit{new link capacity - old link capacity}}{\textit{new link cost - old link cost}}$$

$\hat{M}_1$ = normalized value of M_1

M_2 = PL - PL´

where PL = average vc path length before link is added/upgraded

PL´ = average vc path length after link is added/upgraded

$\hat{M}_2$ = normalized value of M_2

The link with the maximum LUI is chosen.

New DELETE-only metric

The *best* link to delete/downgrade is based on its Link Downgrade Index (LDI):

$$LDI = W_1\hat{M}_1 + W_2\hat{M}_2$$

where

W_n is the weight assigned to $\hat{M}_n$ $\qquad 0 \le W_n \le 1$

$$M_1 = (\textit{old link cost - new link cost}) \times \frac{\textit{new capacity - link flow}}{\textit{new capacity}}$$

$\hat{M}_1$ = normalized value of M_1

M_2 = 1- (PL - PL´)

where PL = average vc path length before link is deleted/downgraded

PL´ = average vc path length after link is added/downgraded

$\widehat{M}_2$ = normalized value of M_2

Finally, a special condition has been added to avoid the algorithm's neglect of low cost under utilized links: once the algorithm enters the *exchange* phase special priority is given to links with traffic flows less than a previously defined[7] threshold.

Modification results

Experimentation with the modified heuristic has revealed that the original shortcomings have been eliminated. As expected the LUI and LDI weight values affect the resulting topology (e.g. setting the value of W_1 to 0 and W_2 to 1 results in a fully connected topology). The full implications of altering these weights however, are left to further study.

NDW VERIFICATION

Verification refers to the process of comparing measured performance results with the values calculated by the tool to determine its accuracy in predicting performance measures. If a reasonable correlation exists, it is likely that the tool's design model has captured the primary factors which govern the performance of the particular system. It can then be used with some confidence to evaluate network performance under other sets of conditions (e.g. topology, traffic levels etc).

Figure 7.2 illustrates the steps involved in validating a model [20].

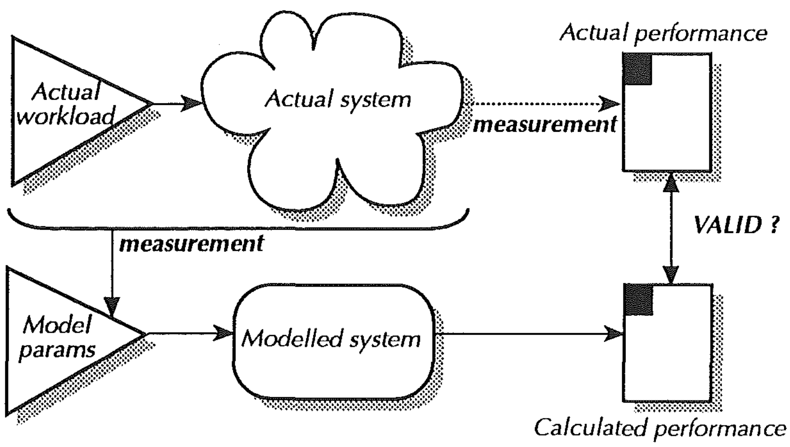

Figure 7.2 Typical validation scheme

7 Refer to pages U-18 and U-19 of Appendix III.

NDW VALIDATION RESULTS

The theory of queues provides a solid mathematical framework for the formulation of these models. In terms of accuracy, a large body of experience indicates that queueing network models can be expected to be accurate to within 5 – 10% for utilizations and throughput, and 10 – 30% for delays [20]. This level of accuracy is consistent with the design requirements of this study.

It is important to ensure that the accuracy of the performance model is consistent with the levels of accuracy available in the other components of the system analysis process (e.g. workload characterization). The essential fact remains — it is pointless to try to improve the accuracy of any network design model by say 3%, if the traffic data feeding that model is 25% off reality.

Traffic flows

The following tables compare the results gained by NDW with the measured performance data extracted from the logfiles.

Node	NDW (%)	Measured (%)
Epsom	29	32
Grafton	21	23
Wellington	15	17
Christchurch	10	8
Hamilton	8	10
Palmerston North	9	8
Lower Hutt	8	8
Timaru	3	3
Napier	7	6
Tamaki	6	7
Auckland	10	13
Rotorua	6	5
Dunedin	5	6
Tauranga	2	2
Invercargill	5	6
New Plymouth	2	2

Table 7.a Node utilizations

Link	NDW (PPS)	Measured (PPS)
EPS to TKI	1.0	0.53
EPS to GFT	12.0	10.1
EPS to WEL	33.0	30.2
EPS to NPL	3.0	2.5
GFT to AKL	16.0	19.0
WEL to PMR	19.0	16.0
WEL to CHC	16.0	16.0
WEL to AKL	12.0	15.4
WEL to LHT	8.1	8.3
WEL to NAP	6.0	5.0
CHC to TIU	2.0	1.5
CHC to DUD	11.1	11.1
CHC to INV	6.0	5.9
CHC to AKL	12.0	11.3
HLZ to AKL	17.0	19.0
HLZ to ROT	9.2	10.9
HLZ to TRG	3.0	2.3
HLZ to PMR	5.0	3.5
PMR to LHT	0.9	0.1
NAP to ROT	1.0	0.3
TKI to AKL	3.5	4.8
DUD to INV	0	0.01

Table 7.b Link flows

Response times

A series of network response time measurements were conducted to test the accuracy of the NDW model. Three on-line *loop-back* tests[8] were performed and compared with the corresponding response time[9] calculated by NDW.

- Test one: PC—PAD—Epsom node—PAD

  NDW: 362ms

  Measured result range: 280-330ms

- Test two: PC—PAD—Epsom node—Wellington node—PAD

  NDW: 433ms

  Measured result range: 302-380ms

- Test three: PC—PAD—Epsom—Wellington—Christchurch—Dunedin—PAD

  NDW: 602ms

  Measured result range: 390-610ms

---

[8] A program was developed to measure the elapsed time between the sending of a packet to a specified destination and the receipt of an echoed reply.

[9] NDW's response time command is detailed on page U-16 of Appendix III.

Conclusions

The measured results for node and link utilizations are within 0 to 25% of the NDW model. It is clear that this variance is higher than the underlying theory should predict. It is suspected that this variation can be attributed to inaccuracies in the COI traffic matrix. The response times estimated by NDW are within 15 to 20% of the measured values.

These performance comparisons indicate the NDW model is in reasonably close agreement with the logfile measurements for medium network loads. The ability to generate performance data on the basis of the available COI information, validate this information, and then improve the results by refining the COI matrix is one of the more useful characteristics of queueing network models. In this particular study the results presented in this section were considered adequate for the design task being addressed.

NDW DESIGN ISSUES

USER INTERFACE DEVELOPMENT

During NDW's one year developmental period many features were added to increase the tool's effectiveness — it soon became obvious that "user interfaces can always be improved." The development of an effective interface is an iterative process that may require several generations to provide a truly effective system.

NDW's user interface evolved over four major generations, with each generation involving numerous small changes. There is no doubt that the on-going development of NDW's interface will benefit from the use and feedback from the user community.

SOFTWARE TRADEOFFS

One of NDW's primary design requirements was to maintain high levels of user-tool interaction. This made it necessary to maintain a tightly coupled feedback loop between user actions and the tool's responses. To achieve this, the software tradeoff between memory requirements, method complexity, and speed were constantly kept in mind. Ultimately the speed of execution took priority over the other constraints.

The design of NDW's data structures were of supreme importance. For small design projects (1 to 20 nodes) the software tradeoffs are not too important. But for larger structures (50 to 100 nodes), the implementation techniques could mean the difference in software requirements ranging from linear to exponential.

The seventeen primary data objects maintained by NDW had to be divided into three categories — those which required instant random access (i.e. the network description matrix), those which required instant sequential access (i.e. the subscriber descriptions) and those which required only delayed sequential access (i.e. user preference settings).

Structures that required sequential access were stored in space efficient but slow link lists. Structures that required instant random access were stored in fast but space inefficient arrays. Finally, the data objects that were accessed only occasionally were kept in scratch files on disk.

Because the Network Description Matrix (NDM) would be scanned hundreds of times during the normal course of operation the following observation proved very valuable — The NDM is symmetrical, that is NDM[i,j] = NDM[j,i] so only half of the matrix has to be scanned thus:

```
for (source:=0; source<max_node_num-1;source++)
   for (dest:=source+1; dest<max_node_num; dest++)
```

In this way processing time and memory requirements were minimized.

Further discussions on the software tradeoffs involved in the implementation of a design tool can be found in [14] and [95].

SECTION IV

ANALYSIS OF THE EXISTING NETWORK

I've measured it from side to side:
Tis three feet long and two feet wide.
William Wordsworth 1770-1850
The Thorn (1798)

chapter eight

THE HUGHES IPN ENVIRONMENT

NET-X is a multi-million dollar wide area data communications network which is owned and operated by NETWAY Communications Ltd. NET-X provides an economical X.25 packet switched service for companies distributed around New Zealand.

This chapter introduces the characteristics and operational background of NET-X including its current topology, architecture, and network monitoring facilities.

The latter sections of this chapter describe the practical methods and software tools developed during this study which enabled the extraction of traffic data from the NET-X network control system.

NETWORK STRUCTURE

The NET-X packet switching network is based on the Hughes Network Systems Integrated Packet Network (IPN) technology. The network currently consists of sixteen nodes (or switches) geographically dispersed around New Zealand (see figure 8.1). The nodes are interconnected by 22 digital backbone links[1] ranging in speed from 9600bps to 64000bps.

NET-X currently interconnects 263 subscribers, including 67 host computers, 3 network gateways and 193 terminal centres using an MCMS topology.

Figures 8.1 and 8.2 detail the geographical and logical topologies as of 12th June 1990. Forty-five of the sixty seven hosts are connected to one of the four nodes in the Auckland city area namely: Epsom, Grafton, Tamaki, and Auckland. The concentration of hosts in this small area would later influence the final network design.

[1] Leased from Telecom.

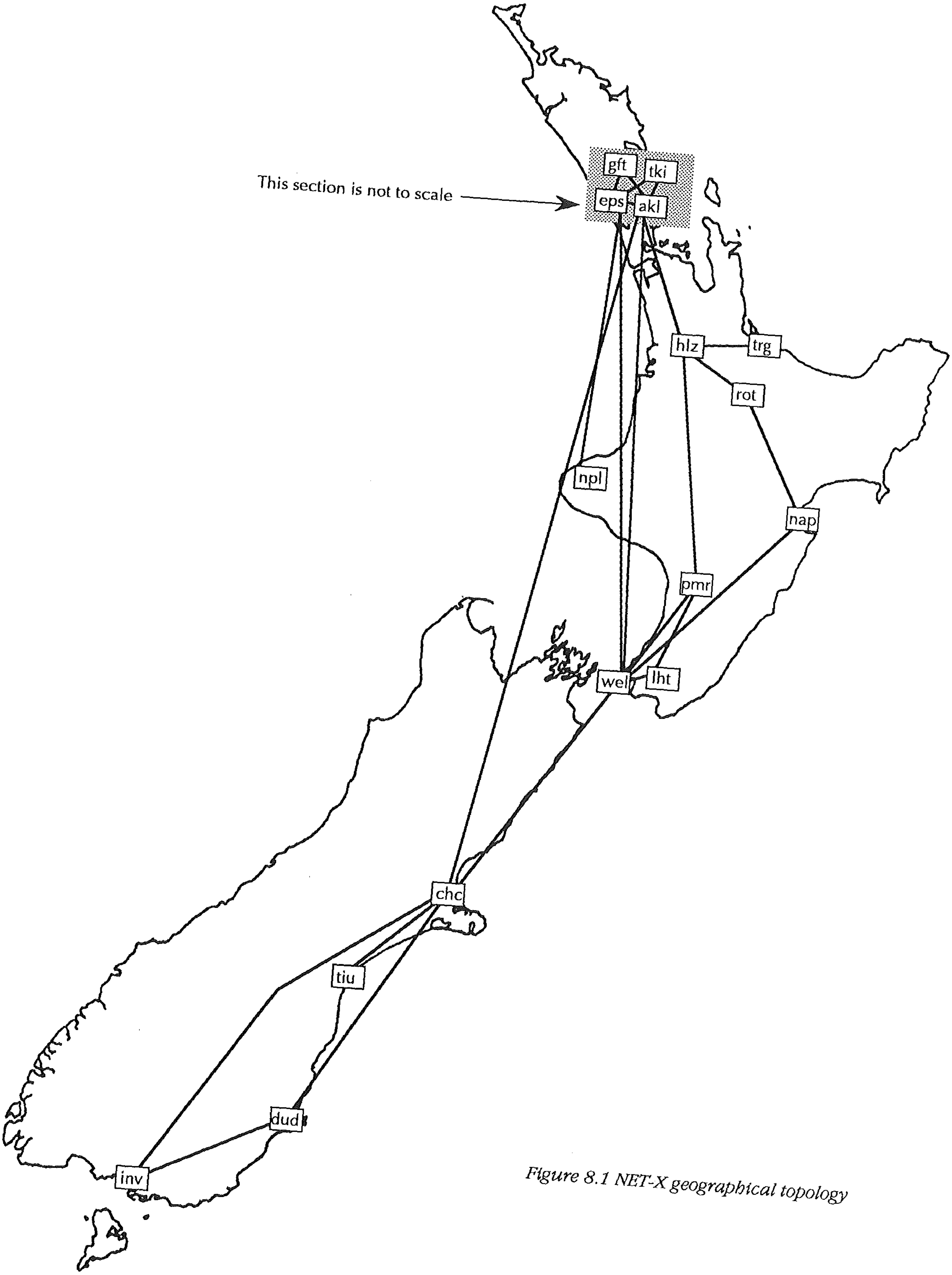

Figure 8.1 NET-X geographical topology

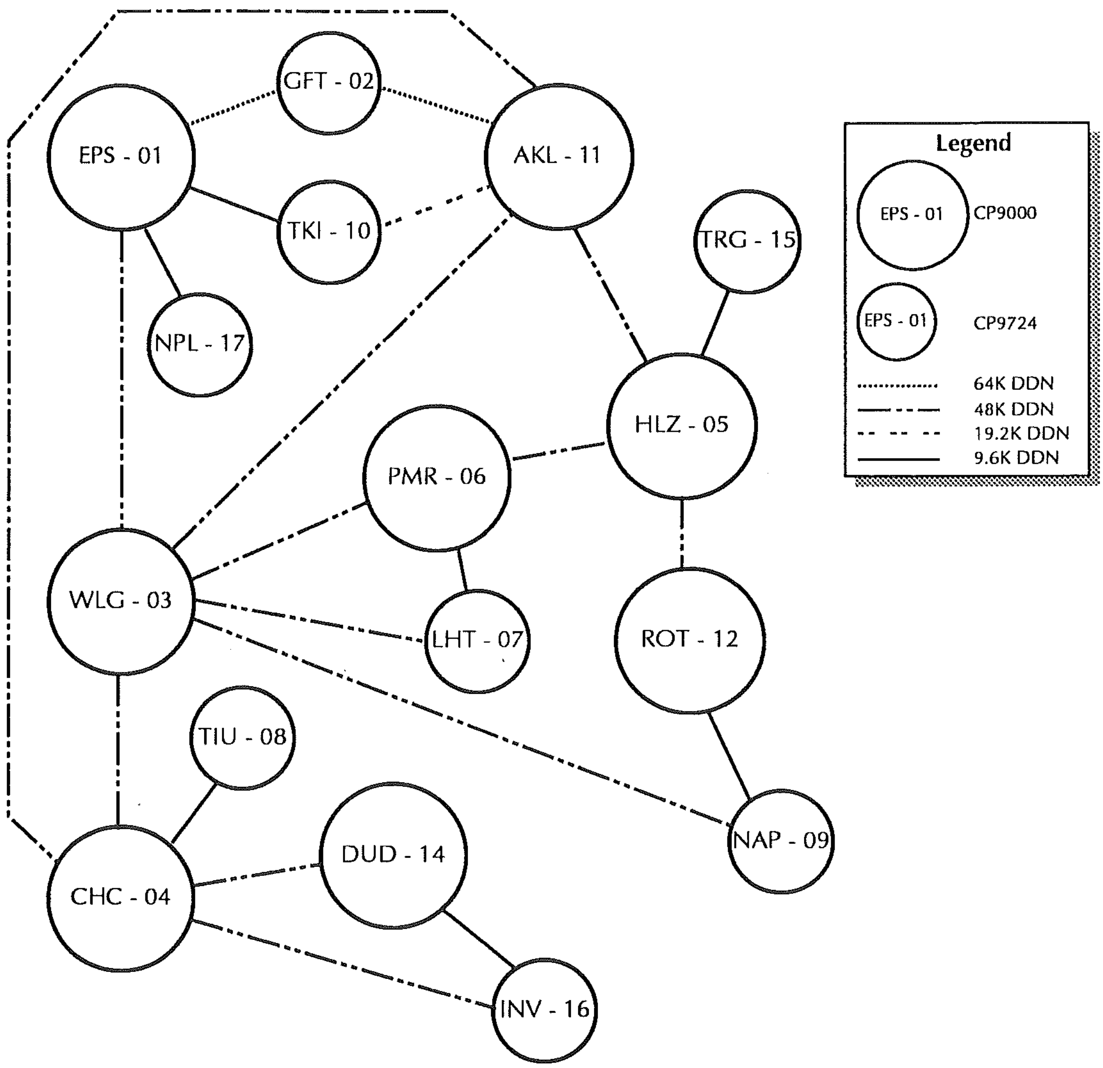

Figure 8.2 NET-X logical topology

ARCHITECTURAL OVERVIEW

The Hughes IPN is a full service X.25 private packet switching network designed and manufactured by Hughes Network Systems, a subsidiary of the Hughes Aircraft company. Many large public (e.g. GTE Telenet, RCA Cyclix) and private (e.g. Ford Motor Co., US Air Force) US packet switching systems are based on the Hughes IPN technology [43].

THE HUGHES IPN SYSTEM

The Hughes Integrated Packet Network system is representative of current generation packet switching systems, consisting of nodes based on modular architectures coupled with elaborate monitoring and control facilities.

System components

The Hughes IPN consists of the following primary elements:

❑ *The CP 9000 Packet Exchange Family* — The CP 9000 packet exchange family represents a series of highly modular and expandable packet switching nodes. Each node consists of one or more Packet Switching Clusters (PSC) which communicate over a common high speed (10Mbps[2]) serial Message Interchange Bus (MIB).

Each PSC consists of a Processor Module (PM) and up to three Line Interface Modules (LIMs). The PM provides the cluster with a general processing capacity of approximately 202 Packets Per Second (PPS).

The LIMs are connected to the cluster's PM through a parallel Multi-Master Bus (MMB). Each LIM provides processing for up to eight serial communication ports.

All user access and backbone links are connected to one of the eight ports on the LIM's I/O module.

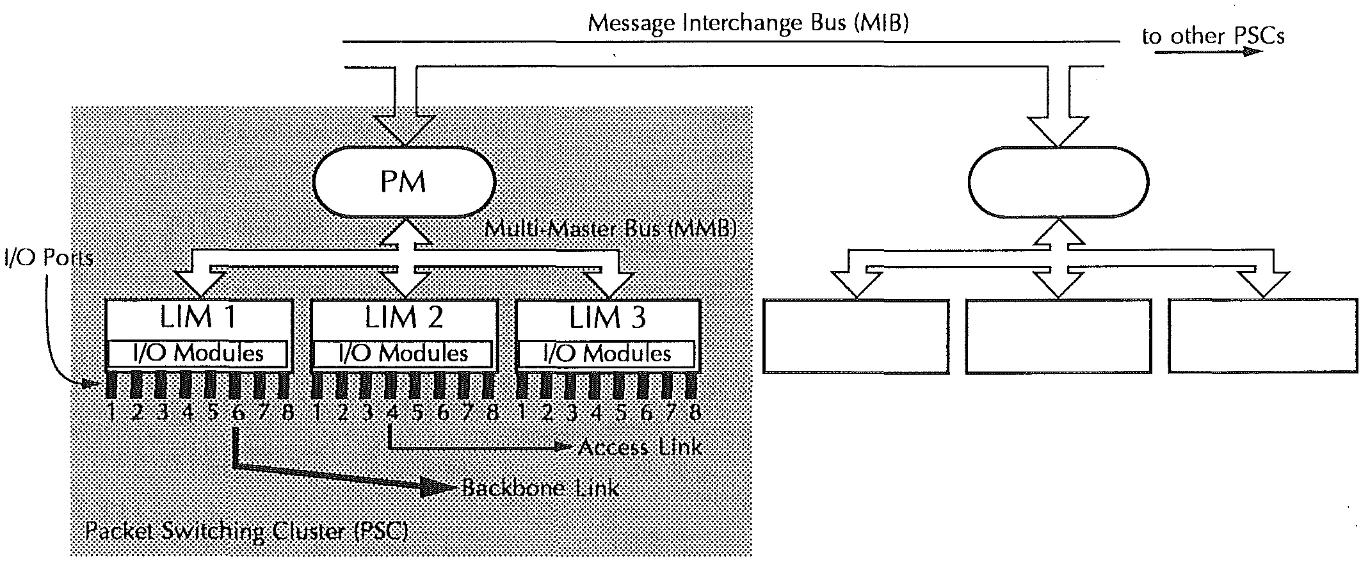

Figure 8.3 Hughes IPN node architecture

❑ *The Network Control System* — The Network Control System (NCS) offers advanced network managing facilities and represents the overall 'intelligence' of the system. The NCS is administered by a Network Control Processor (NCP)[3] which is connected to the network through multiple X.25 links. The NCP is primarily responsible for the handling of the administrative and network service functions (e.g. the storage of operational statistics, network component (re)configuration, the validation of call addresses etc.)

---

[2] 3000-4000 PPS @ 50% utilization [41].

[3] The NCP is implemented on a DEC MicroVAX minicomputer. In larger network configurations two or more NCPs are usually installed.

The network operator interacts with the NCP by way of a dedicated Network Operator Console (NOC). The operator can use the NOC to perform sophisticated on-line administrative functions including:

- The configuration and control of network components (e.g. restarts, switch overs, software updates etc.);
- The addition of network components, subscribers etc;
- The direct monitoring of network components for utilizations, errors etc.

Figure 8.4 illustrates the interaction between the NCS components.

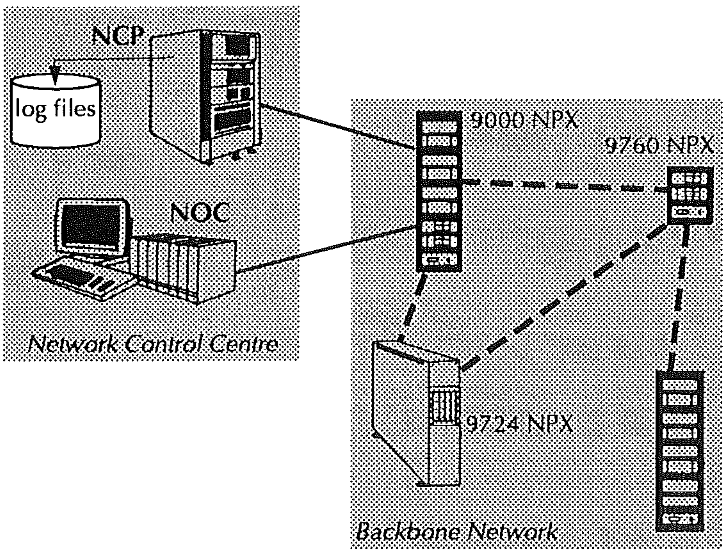

Figure 8.4 IPN network control system

NETWORK REPORTING SYSTEM

Central to the traffic measurement goal was the IPN's statistics collection facility. The Hughes IPN collects an extensive array of device statistics including node and link utilizations.

Each backbone node monitors itself and stores performance information on its own local hard disk. The node then periodically[4] forwards this information to the NCP for further processing. Once the node has sent the statistics it resets its statistical counters and begins recording for the next period. The NCP on receipt of this information stores it in one of the five separate log files it maintains.

Only the call record and statistics logs maintained by the NCP were of interest to this study.

---

[4] The data collection interval can be configured with a minimum granularity of one hour [42].

Call Records log: crmlog.dat

Most networks that share resources provide some mechanism for billing individual customers for the use of these resources. The chief method used in packet networks to accomplish this is through the analysis of call records.

The IPN stores comprehensive information on every virtual circuit that is established within the network. Typically, this file contains 20-30,000 records detailing a twenty-four hour period. Most of these records however, can be attributed to system related and duplicated call verification details.

Statistics log: sttlog.dat

Each component within the Hughes IPN environment (i.e. PMs, PSCs, LIMs etc.) periodically generates a statistics record and forwards it to the NCP. The NCP collects these records and stores them in the statistics log file.

The information stored in the *sttlog.dat* log can be categorized into three primary groups:

- Link statistics (e.g. packet counts, number of VCs established etc.);
- Resource use statistics (e.g. Buffer usage, CPU usage etc.);
- Performance statistics (e.g. Component throughput etc.)

For a complete list, the reader is referred to [40].

DATA COLLECTION TOOLS

The NET-X environment had no facilities for the archiving and analysis of the statistics logs once they had been forward to the NCP. Hughes do market a Network Management Report (NMR) package which was unfortunately unavailable at the time of the study[5]. Consequently several C programs [51] were developed to extract and collate the logfile information into a series of tabular reports. These reporting tools have the same functional capability as the NMR package [44] and produce nine different reports on network activity. An overview of these reports can be found in Appendix IV.

The reports are also produced in a format which can be directly read by any microcomputer based graphing program. This section outlines the tools and methodologies used for the collection of this essential data.

LOGFILE CAPTURE AND ARCHIVAL

The log files maintained by the Network Control Processor are circular in nature. In other words, at the end of each twenty-four hour period, the NCP starts recording the next period over the top of the old file.

---

[5] The NMR package was prohibitively expensive at a cost of $US18,300.

To capture copies of each twenty-four hour period, a Digital Command Language (DCL) script running under Digital's VMS operating system on the NCP was developed. This script automatically archives the log files at the end of every day. The script activates at 12:45am (allowing forty-five minutes for the devices to send their reports for the 11:00-12:00pm period) and copies the logs to separate files each with unique names based on their archival date.

Each logfile consists of approximately six megabytes of binary data.

ANALYSIS OF THE CALL RECORD LOGS

NET-X records the details of every virtual circuit set up within the network. The details of each virtual circuit are forwarded and stored as 160 byte binary records within the circular *crmlog.dat* file.

A program was developed for the off-line analysis of this call record file. The program scans the file, discards all duplicate[6] and invalid[7] records, and generates reports based on the remaining data.

This program extracted a myriad of information, including:

- Distribution of the virtual circuit routes (see fig 9.6);
- Virtual circuit path lengths (see fig 9.7).

In addition, each call record contains the virtual circuit's source and destination address. This enabled the determination of source-destination pairs within the network and the subsequent construction of NDW's Community Of Interest (COI) traffic matrix.

ANALYSIS OF THE DEVICE STATISTICS LOGS

The network device statistics are stored as 100-200 byte[8] binary records within the circular *sttlog.dat* file.

A second program was developed to perform the *sttlog.dat* file analysis. In a similar fashion to the call records analysis, all duplicate and invalid records are discarded. The program then accumulates information on each network component from the remaining log file data. The user can then query the program to produce the reports detailed in Appendix IV.

The statistics logs contained the most essential data needed for the design tool, including submitted traffic rates and packet size distributions (see figure 9.8)

---

[6] Because the NET-X has only one Network Control Processor every call record is duplicated.

[7] 3 to 4 % of the records within the file contain invalid data (e.g nonexistent nodes, illegal dates etc.)

[8] The size varies according to the type of statistics record.

CONCLUDING REMARKS

Two network reporting programs have been developed during this study which have enabled the off-line processing of the NCP log files to provide reports on the IPN activity. Reports may be generated for the analysis of network utilization, for the occurrence of errors in the network, and for profiling the call traffic among network users.

The analysis programs discussed in this chapter have provided the NET-X operators with a data collection and analysis facility which has proven invaluable for gaining insight into the network's operational state.

The results of the analysis of network data over a one week period are discussed in the next chapter.

NETWORK WORKLOAD CHARACTERIZATION

To engineer an existing computer network successfully, periodic measurements must be taken. The primary purpose of the measurement effort was twofold. First, to provide some insight into the current state of the network— to observe the traffic characteristics of the operating network and to highlight possible areas where redesign may be necessary. Second, to provide accurate and representative input data for Network Designers Workshop.

Obviously, the quality of results from any design effort is directly related to the accuracy of the input data. It is therefore essential the engineer builds up an accurate understanding of the current network state before any design work is embarked upon. The set of measurement tools discussed in the previous chapter have proved invaluable for providing insight into the network's usage and behaviour.

This chapter will discuss how the measurement tools have been selected and co-ordinated to evaluate the network's operational status and details some of the results gained by them.

THE MEASUREMENT PROCESS

The activities involved in the characterization of network workloads can be split into the four processes detailed in figure 9.1 [11].

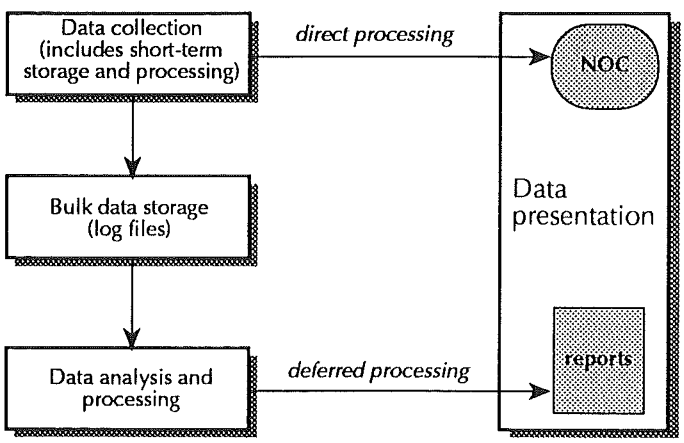

Figure 9.1 Measurement processes

Measurement data consists primarily of event counts and traffic intensities on network resources. The processes involved in capturing this information are as follows:

- *Data collection* — the collection and short-term processing of event, traffic, and call record information. Measurement data is either archived for deferred processing by off-line analysis programs or directly processed by on-line monitors (such as the Hughes NOC). The measurement granularity of data processed at this stage is usually much higher than the archived data.
- *Bulk data storage* — the secondary storage of the accumulated traffic data as log files.
- *Data analysis and processing* — the processing and collation of the traffic data archives.
- *Data presentation* — the process of translating the measurement data into readable form.

TRAFFIC MEASUREMENTS

The network planning process requires information on the intensity and behaviour of traffic within a network system. Ideally, traffic statistics for network planning should be collected for at leased sixty days (not necessarily consecutive) over the previous twelve months in which the mean busy hour is the highest [12]. Owing to time constraints, only two weeks of data was collected for this study.

Network traffic data was collected over a continuous fourteen day period from 12:00pm on the 28th July 1990 to 12:00pm on the 11th August 1990. The traffic data consisted of 28 log file archives totalling 172 Megabytes of binary data. The analysis tools discussed in Chapter eight were used for the deferred processing of this two week snapshot. The reports generated by these programs gave rise to the following results.

TRAFFIC INTENSITY

The analysis of the traffic reports revealed distinct traffic patterns that were dependent on the selected time frame and would tend to vary in step with the network user community. These variations were investigated and the peak periods determined.

Variations during the week

Figure 9.2 depicts the total daily traffic variation throughout the measured week. Weekly traffic forms a regular pattern— daily peaks exist with greater traffic volumes on Mondays and Fridays and substantially less during the weekend.

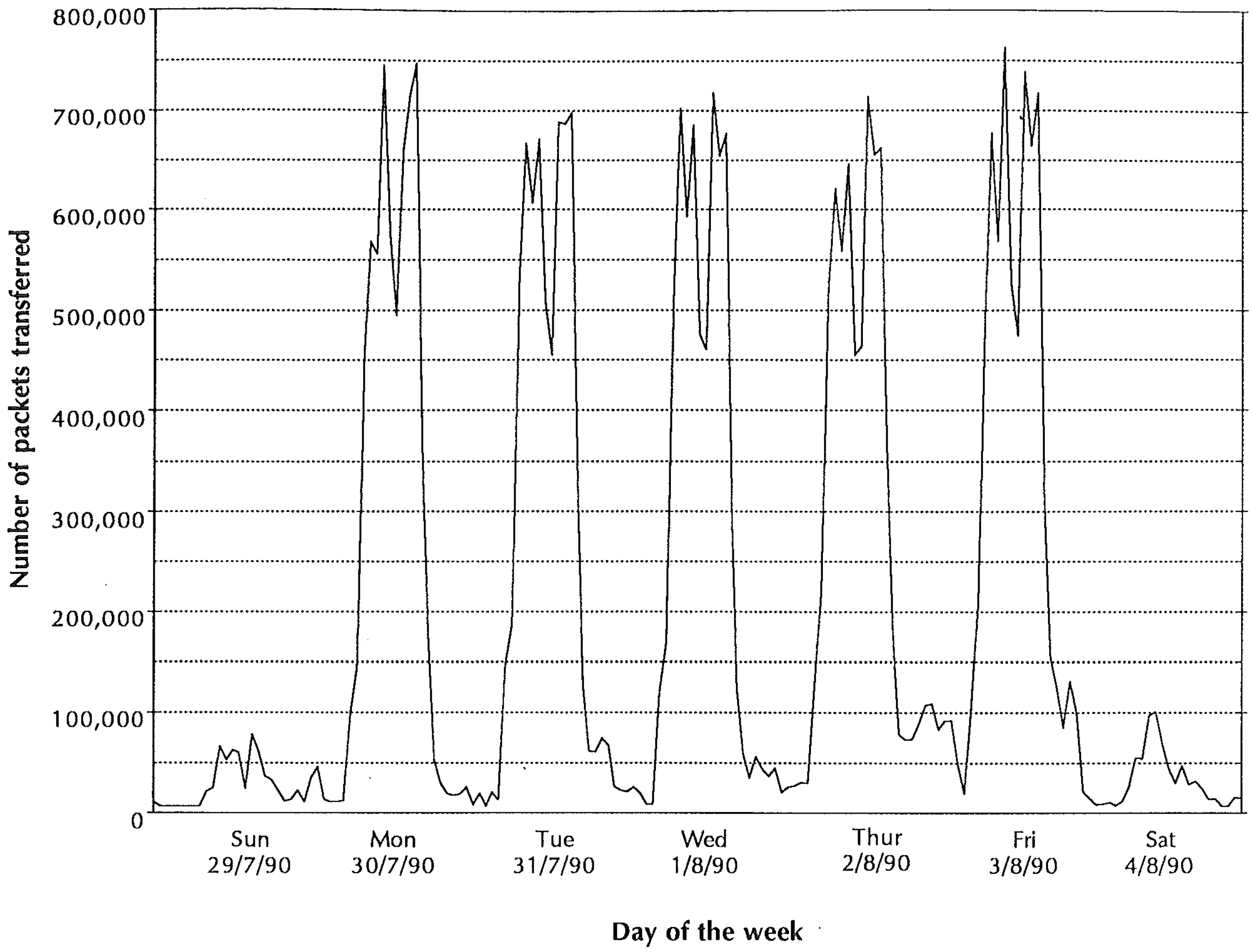

Figure 9.2 Total daily network traffic for the selected week

Variations during the day

Generally, traffic will peak two or three times during a normal business day [47]. The node cluster reports generated statistics that enabled the identification of the peak hour traffic periods. Figure 9.3 illustrates the averaged Grafton node traffic intensities for Monday 30th July 1990.

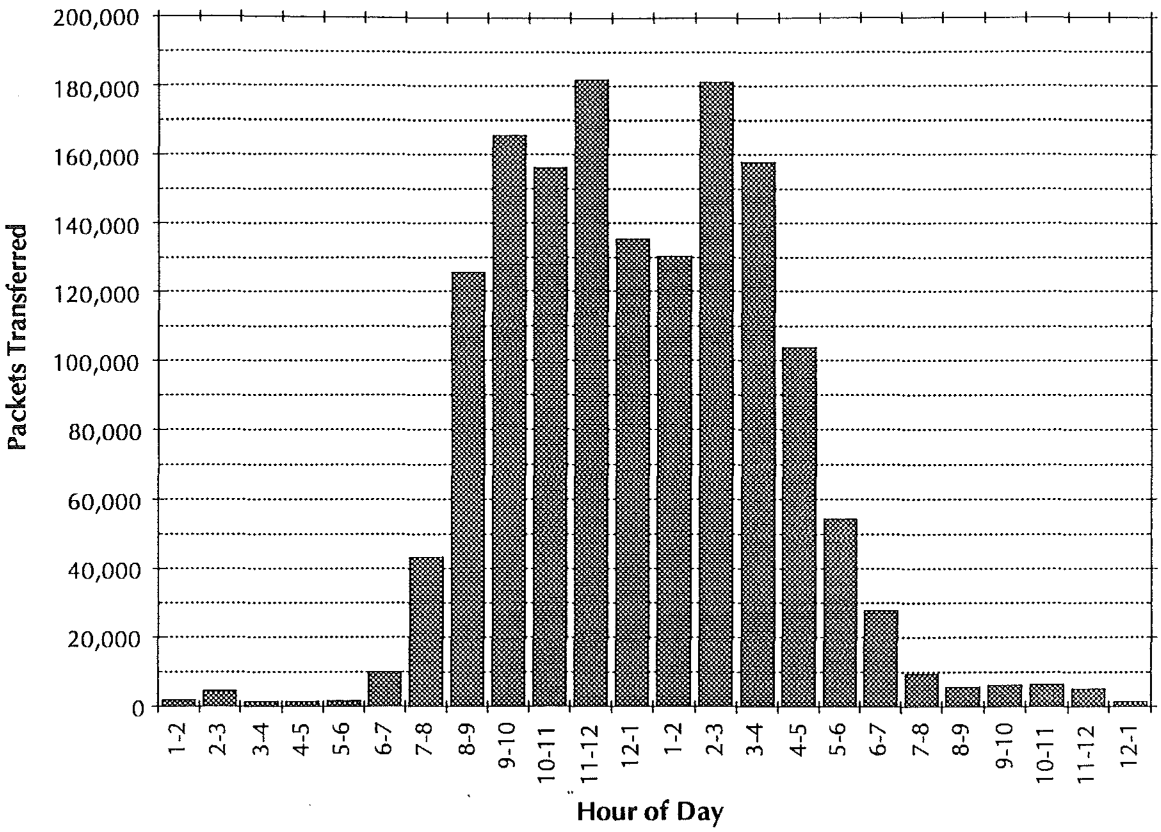

Figure 9.3 Node traffic profile for an average business day

As one would expect, traffic intensities are closely correlated to business "time of day" patterns. Two distinct traffic peaks exist; the first occurs between 11:00–12:00 noon and the second between 2:00–3:00pm each day. Approximately 20 to 25% of the days traffic is transmitted during these periods.

Analysis of the daily traffic intensities on other network components all result in similar usage patterns — first, a rapid increase in traffic from 6:00am to 10:00am and then a slight decrease[1] soon afterward. Traffic intensities increase to reach the first peak around 12:00 noon and drop off during the lunch hours. Traffic flows gradually build up to the second peak at around 3:00pm, then drop off sharply after 4:00pm. Traffic flows outside office hours are almost non-existent —those that do exist can be attributed primarily to HNS supervisory traffic[2].

Variations during the hour

The following graph illustrates the one minute variations in traffic intensity for a communication link over an hourly period. Because the Hughes NCP will log statistics with a maximum granularity of only one hour [42] the graph in figure 9.4

---

[1] Morning tea break?

[2] Devices forwarding statistics to the Network Control Processor (NCP).

is based on *directly processed* measurements[3] taken from the Network Operators Console.

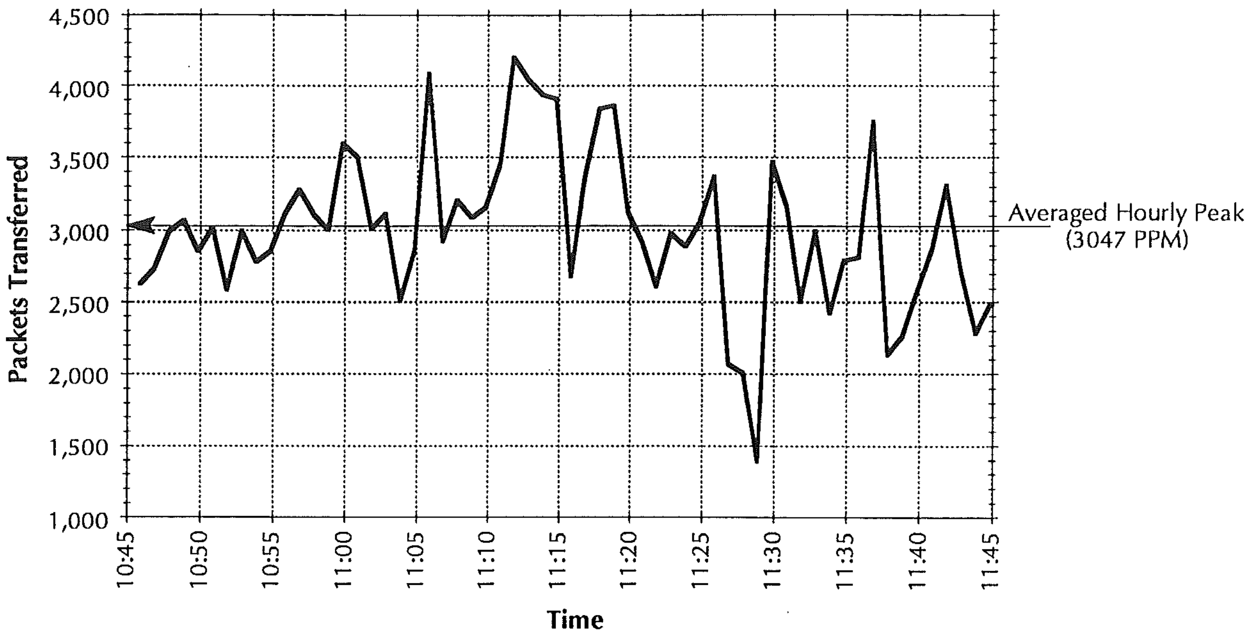

Figure 9.4 Traffic fluctuations for an average hour

Analysis of traffic variation within an hourly period revealed no regular patterns and can therefore be considered as purely random [82].

TRAFFIC SAMPLING ISSUES

The selection of the traffic sampling time frame[4] resolution represents a tradeoff between the data collection cost and the accuracy of the traffic peak measurements. If the sample periods are too widely spaced the averaged peaks may not adequately represent the transient effects sought, while sample periods too closely spaced may require excessive data collection and will consequently overtax the network management system.

When determining peak traffic rates it is important to note the period over which it is calculated. For example, averaging the hourly peak traffic rate will tend to underestimate the sub-hourly peaks within that time frame. The exact magnitude of this deviation is dependent on the nature of the traffic during that hour[5]. Sometimes the one minute peak can be 10 to 20 times the hourly average [65].

Table 9.a illustrates the averaged peak values for varying time frames and the resulting magnitude of deviation from the peak one minute flow.

---

[3] Refer to figure 9.1.

[4] Most modern network management systems allow the designer to specify the monitoring granularity.

[5] For example, file transfers would result in a fairly stable pattern whereas some user traffic can result in 'lumpy' traffic patterns.

Time frame (min)	Averaged peak (PPM)	Deviation (%)
60	3047.30	38.00
30	3318.23	26.72
15	3399.33	23.70
5	3910.40	7.53
1	4205.30	0.00

Table 9.a Variation of averaged traffic peaks with measured time frame

The network designer must decide which traffic peak time frame to use when sizing the network. If a system is designed to accommodate the averaged sixty minute peaks, then the performance of a component may be severely degraded during the one minute peaks. Similarly, if the system is designed to accommodate the one minute peaks, then components will be somewhat under-utilized at other times of the day. The goal is to design a system that achieves a desired balance between average utilization and acceptable levels of congestion under peak loading.

Figure 9.5 illustrates how the average packet delay within NET-X changes with the total (relative) load imposed on the system[6].

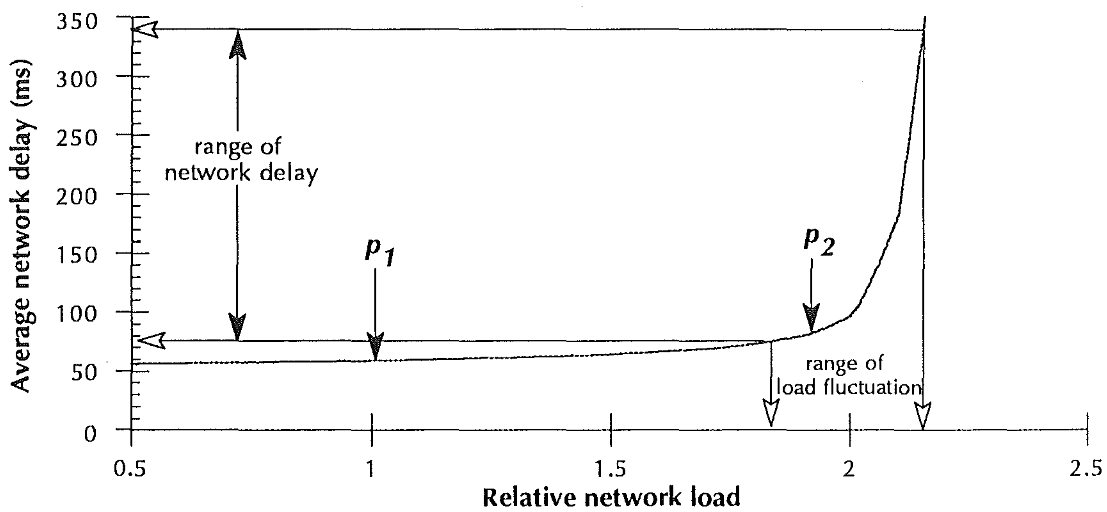

Figure 9.5 Network delay fluctuations caused by changes in arrival pattern.

Point P_1 on the graph represents the current load on the system using the averaged hourly utilization figures. This load results in an average packet delay of 58 ms.

---

[6] Calculated using NDW's sweep function (see page U-20 of appendix III for more details).

One can see that minute by minute variations in traffic intensity (of the same magnitude detailed in Table 9.a) would cause the delay to fluctuate between 56 ms and 61 ms[7]. If however the averaged hourly load corresponded to point P_2 on the graph then the sub-hourly traffic fluctuations would result in delays ranging from 64 ms to 340 ms. From the user's point of view, the performance of the system would appear quite erratic. It is therefore important that the designer knows at which point on the curve the network is operating before any decisions concerning the Community Of Interest (COI) traffic values are made.

The design detailed in Chapter ten is based on the averaged hourly traffic peak figures. This is acceptable because NET-X is operating in a stable region of the curve (i.e. point P_1).

TRAFFIC DISTRIBUTIONS

In reality, traffic is not uniformly distributed over the network. Many network design theories make no allowance for the traffic imbalances that arise within cities and customer applications [66]. These traffic imbalances can be significant and may require the extra deployment of network switches and links to accommodate the heavier demand in such areas.

The point-to-point traffic loads are vital input for the network design tool and are usually described using a COI traffic matrix — where row and column headings represent source and destination locations and their corresponding cells providing quantification of the traffic flowing between them. There were three possible approaches for obtaining the data for the COI traffic matrix:

- *Direct measurement of point to point traffic* — This approach was impractical because of the huge investment in time and equipment needed for collecting such data.
- *Estimation from available measurements (backbone loads etc.)* — Methods for estimating point-to-point traffic loads from the available backbone load data have been proposed by Wang [97]. The accuracy of Wang's method however is highly dependent on the network topology.
- *Extraction from the client call records* — This approach was adopted by this study primarily because it provided an accurate and facile method of extracting the COI data. The call record analysis programs discussed in Chapter eight enabled the determination of the COI matrix, along with the following results.

Virtual circuit route distributions

Analysis of the call record files revealed that 72% of the total traffic either originates or terminates at the Epsom, Auckland, and Grafton nodes. The COI also indicated that approximately 27% of all virtual circuits set up are local to the

[7] The fluctuation projection arrows for point P_1 have been omitted for clarity.

Epsom and Auckland Nodes (see figure 9.6). This result would later reinforce the importance of maintaining a *cluster* of two or three nodes within the Auckland area in any future designs.

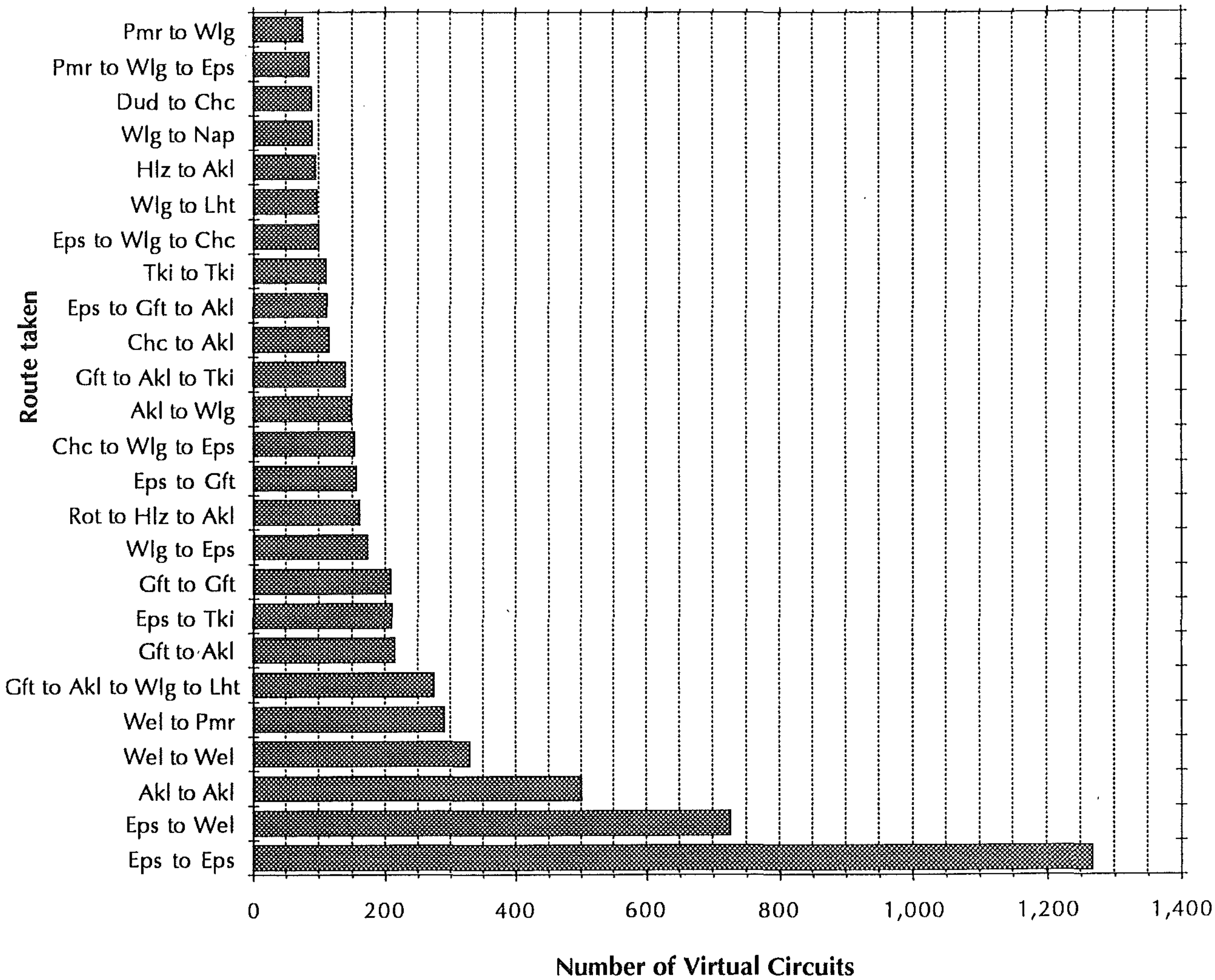

Figure 9.6 Distribution of virtual circuit routes

Virtual circuit path length distributions

Figure 9.7 illustrates the distribution of virtual circuit path lengths over a period of one week. Apart from the significant number of virtual circuits with path lengths of 0 (indicating that the call's source and destination are located at the same switch) the distribution is roughly bell shaped.

The shape of this distribution is strongly correlated to the quality of the network topology and it is desirable to keep the distribution peak as far skewed to the left as possible.

The average path length of a virtual circuit indicates several factors:

- *The average delay experienced*— Increasing the path length contributes to packet delays and thus reduces traffic throughput proportional to the number of nodes and trunks involved in the transmission path [84].
- *The network connectivity*— The lower the average path length, the higher the network connectivity. A highly connected network (i.e. no long chains) greatly increases general reliability (and of course cost!).

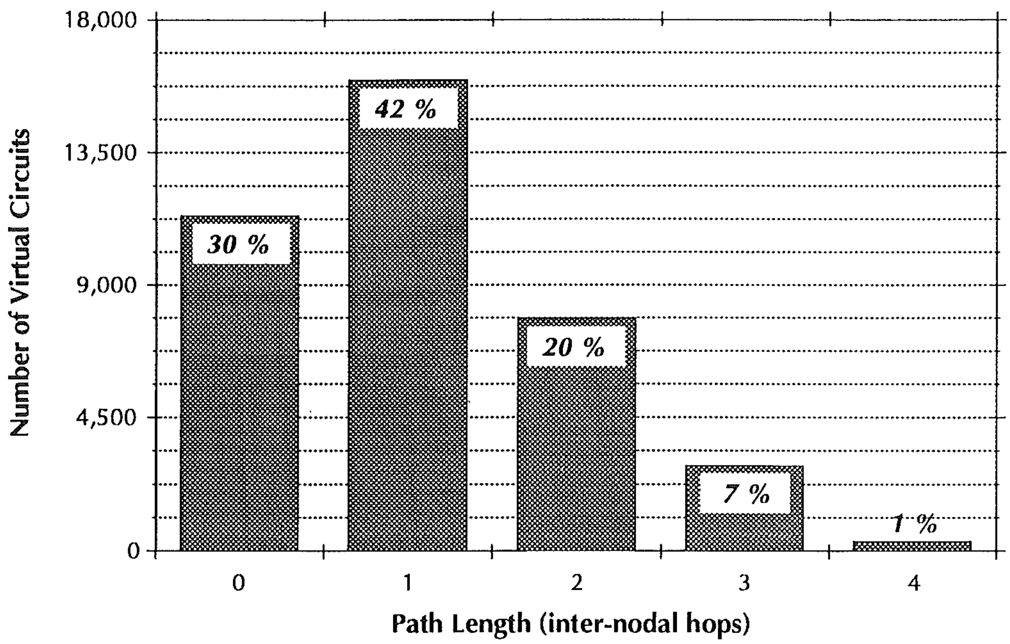

Figure 9.7 Distribution of virtual circuit path lengths

Packet size distribution

Knowledge of the average packet size is essential for any analytically based design tool. Fortunately the Hughes NCS monitors the distribution of packet sizes transmitted over the backbone links. The results of the analysis[8] of several backbone links over a 24 hour period are shown in figure 9.8.

Packet lengths are logged with a granularity of only 16 bytes, so an exact average packet size could not be determined. Using the method of *weighted means* [65] the average packet size was calculated to be approximately 55 bytes long.

[8] Using the packet size summary reports detailed in Appendix IV.

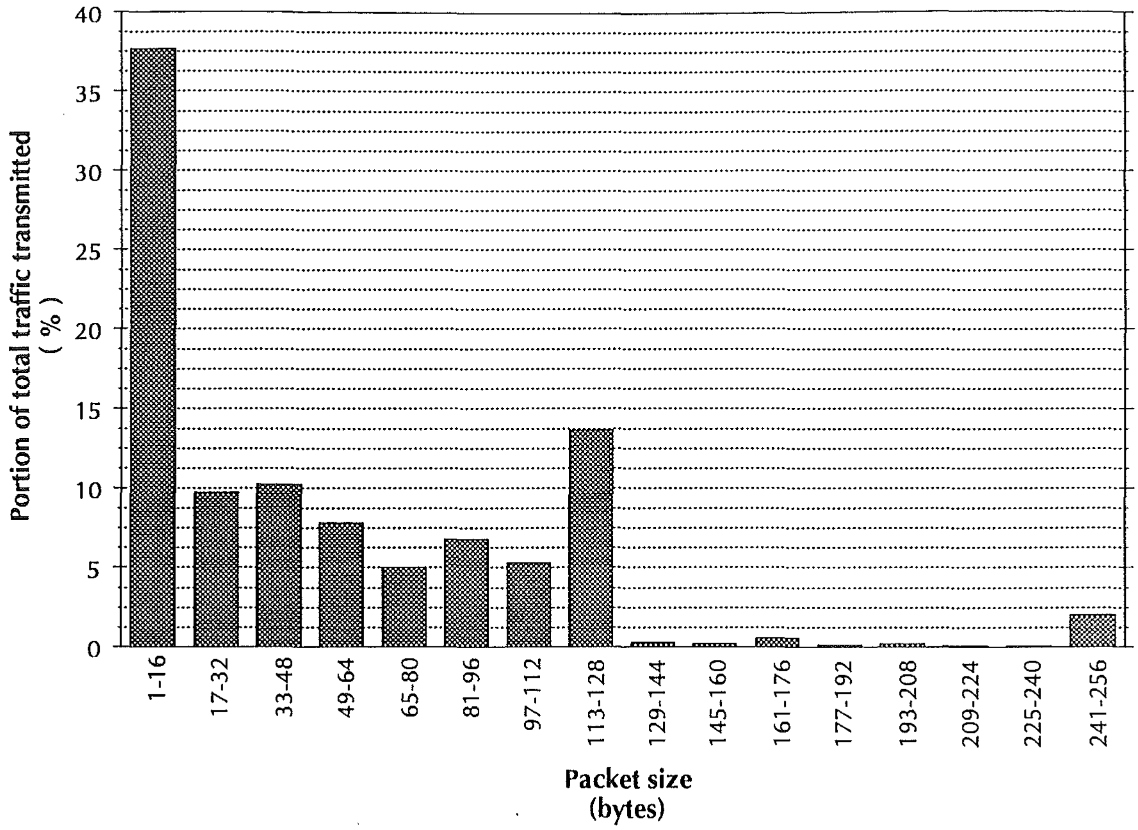

Figure 9.8 Packet size distribution over backbone links

The majority (some 58%) of packets are less than 48 bytes long. Nearly 38% of the total traffic consists of packets between 1 and 16 bytes in length.

The distribution of packet sizes can be attributed to [71]:

- *1-16 bytes* — (1) User applications: Typically, these applications operate in 'block mode' sending fields to the host after they have been entered by the user. These fields are usually 1-40 characters in length. In addition, many applications block and forward user input one character at a time to the host. Consequently a large proportion of packets within this range contain single characters— thus for each character sent eight bytes of X.25 protocol overhead is generated, resulting in a very inefficient use of the network.

  (2) Printer XON/XOFF characters: Flow control characters resulting from remote print jobs.
- *17-112 bytes* — General user traffic
- *113-128 bytes* — Most host to user traffic within the network will be sent as either 128 or 256 byte packets. Consequently significant peaks exist at these two intervals. This first peak is due primarily to print jobs and host computers initiating terminal screen repaints.
- *129-240 bytes* — The packets within this range are NCS log records being forwarded to the NCP.

- *241-256 bytes* — The significant number of 256 byte packets can be attributed to two factors. First, data being spooled to remote printers and second, host to host file transfers.

One can see from figure 9.8 that there are a large number of short packets (1-16 bytes), a number of maximum length packets, and decreasing numbers of packets with intermediate lengths. This distribution of packet lengths is closely correlated to the exponential distribution[9] assumed by the analytical models used in this study.

SHORTCOMINGS

Network design is an engineering rather than a mathematical process. One of the main reasons for this is that the traffic data upon which the process is based is never totally accurate. Several factors contributed to the variation between measured and actual traffic values used in this study.

- *Incompleteness of data collection* — The Hughes NCS logs the traffic data on an hourly basis only [42]. This averaging of the peak traffic intensities can lead to an under-estimation of the actual peak demands on the system.
- *Traffic shifts* — changes in the user profiles will occur between the data collection and design period, and the implementation period. Within the NET-X environment (and indeed all operational systems) "*traffic data grows old very quickly*" [71].
- *Data Recording and processing errors* — Malfunctions in the data collection equipment and data handling errors are unavoidable. For example, closer analysis of the call records used in this study revealed that 3 to 4% contained obvious invalid data (i.e. nonexistent nodes, illegal dates etc). Such errors can be detected and the offending records discarded. Unfortunately, there is no way of detecting and filtering erroneous traffic statistics.

The measurement facilities described in Chapter eight have improved the process of data collection dramatically, however some of the above sources of error will always be present. These shortcomings force the network planning process to remain a tradeoff between desired accuracy and realistic assumptions.

CONCLUDING REMARKS

The measurement and collation of network traffic information constitutes the first and most fundamental part of any network design process. The design phase relies heavily on the accuracy of the input data and without the use of the data collection tools developed during this study much of the final design would have

---

[9] Assumes very small packets are most likely to occur, maximum length packets unlikely, and has no limitation on the packet length.

been the product of mere guess work. The quality of any network design effort will always be directly proportional to the precision of the data feeding it.

The measurement effort detailed in this chapter has successfully achieved the first stage of the design cycle. The following chapter describes the second phase of the design cycle; the coupling of the network traffic data and NDW to optimize the current NET-X configuration.

SECTION V

DESIGN OF THE OPTIMIZED NETWORK

Perfection is reached, not when there is no longer anything to add,
but when there is no longer anything to take away
Antoine de Saint-Exupèry

c h a p t e r   t e n

NETWORK DESIGN WITH NDW

The last two sections of this thesis have introduced Network Designers Workshop, a graphically orientated design tool, and the nationwide packet switching network NET-X.

This chapter describes how the network measurement results have been used in conjunction with NDW for the development of an alternative NET-X structure. A configuration which is relatively insensitive to the dynamics of the user communities demand pattern, offers higher degrees of performance, and is considerably more cost-efficient to run.

This chapter also explores the impact of several design parameters on the optimum cost topology.

DESIGN APPROACH

The design approach employed by this study de-emphasised the importance of finding the lowest cost network design— making that goal secondary to ensuring the final configuration achieved the following:

- The required performance objectives.
- The solution must be cost-effective. Unnecessary nodes and intercity links must be eliminated.
- It must provide a solution that is capable of accommodating growth. The structure must be capable of graceful migration to future topologies.

NETWORK COST TRADEOFF

The hardware, subscriber access, and backbone costs represent the primary cost tradeoff facing the network planner. Figures 10.1 and 10.2 show the interdependence of these three network cost components and how they were found to vary as a function of the number of nodes in the network.

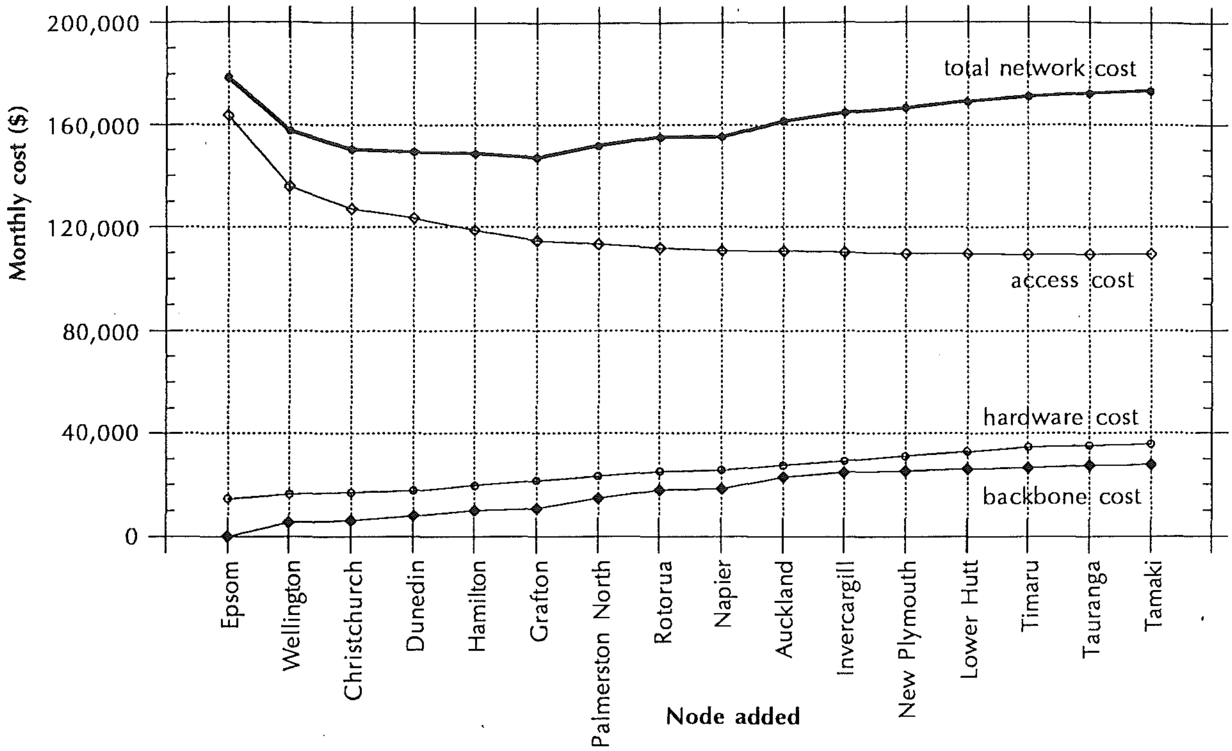

Figure 10.1 Network cost as a function of nodal dispersion (DDS access links)

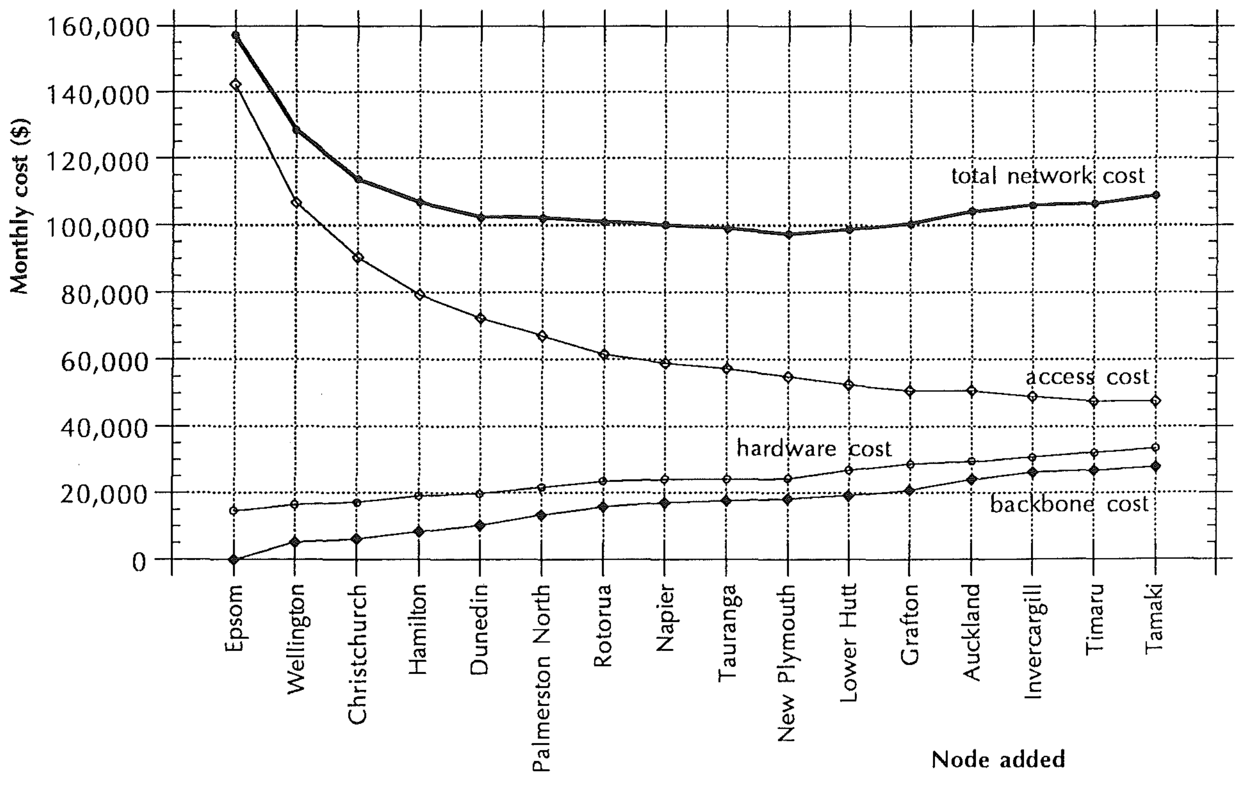

Figure 10.2 Network cost as a function of nodal dispersion (ADS access links)

As previously discussed in Chapter four, there exist three components to the total cost of any two level hierarchical network. These are:

- *Access cost*— The cost of connecting the subscribers to the network. The access costs decrease as the number of nodes increases since there are fewer access link miles. The cost decreases rapidly at first, then gradually, since the access link cost savings incurred by deploying two nodes rather than one is greater than the cost saving incurred by eleven instead of ten.
- *Backbone cost*— The cost of connecting the nodes. These costs increase as the number of nodes increase since there are more nodes to connect.
- *Hardware cost*— The cost of the nodes. This cost increases almost linearly as the number of nodes increase.

Fewer nodes result in reduced hardware and backbone related costs. This is achieved however, with increases in the user access network cost. At some point there will be an optimum tradeoff between the three cost factors. This point is dependent on several factors unique to the network being planned— the primary ones being: the number and geographic locations of the subscriber centres, the cost of the network nodes, and the communication link tariffs.

If a minimal cost structure is to be found, the interdependence of the three network cost components must be taken into consideration. Therefore, it is essential the node placement, access network, and backbone design processes are treated an an integral component. NDW's integrated design approach allows the designer to optimize the total network cost by generating successive network structures[1] until an acceptable configuration is found.

THE NET-X DESIGN PROCESS

This section discusses the cost driven optimization of the NET-X network. The results presented in this section illustrate the applicability of NDW's design method[2] and its ability to offer significantly greater cost savings than non-integrated design techniques.

DESIGN PARAMETERS

Section IV of this thesis details the current network environment used as a basis for the design effort discussed in this chapter. This section summarises those design assumptions.

---

[1] Each structure corresponding to a point on figures 10.1 or 10.2.

[2] As detailed in figure 6.16.

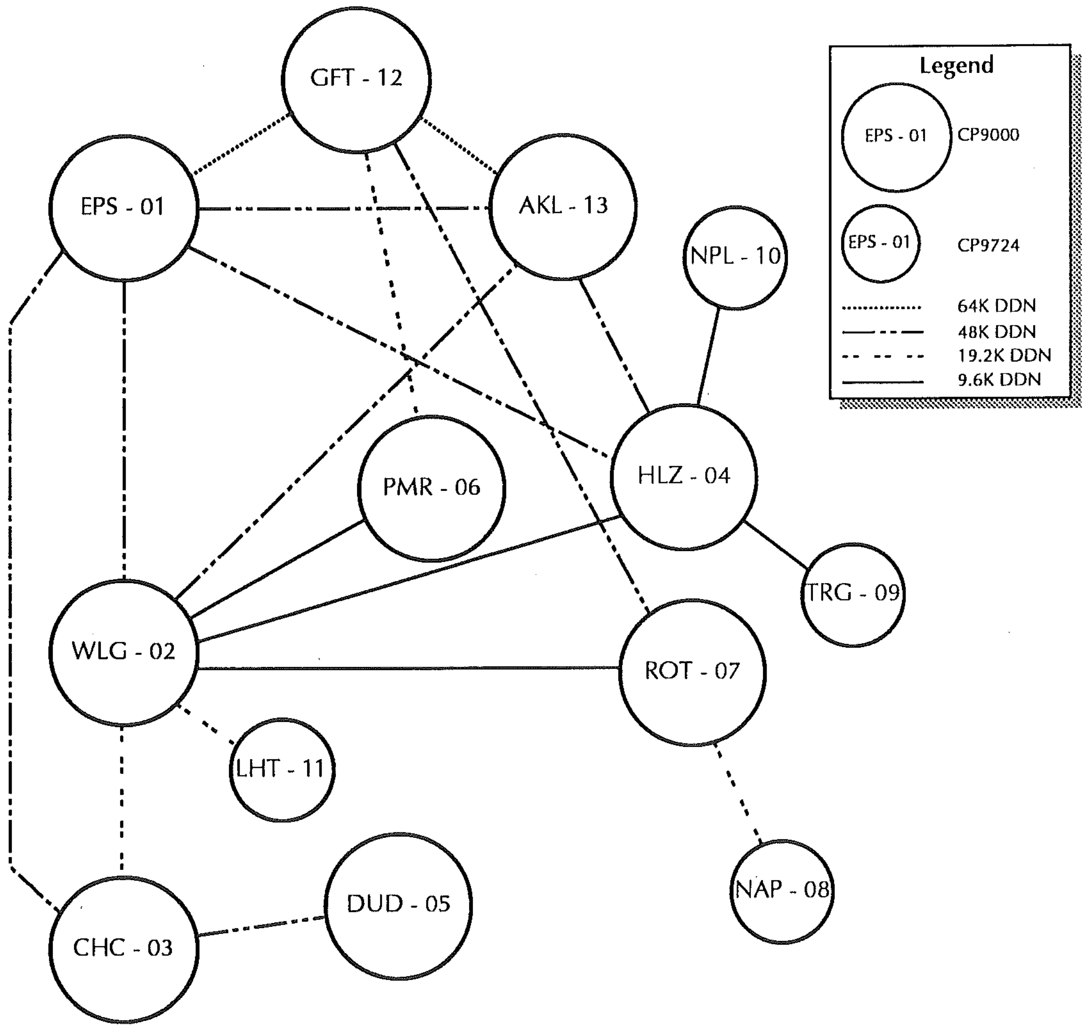

Figure 10.3 Alternative NET-X topology (logical)

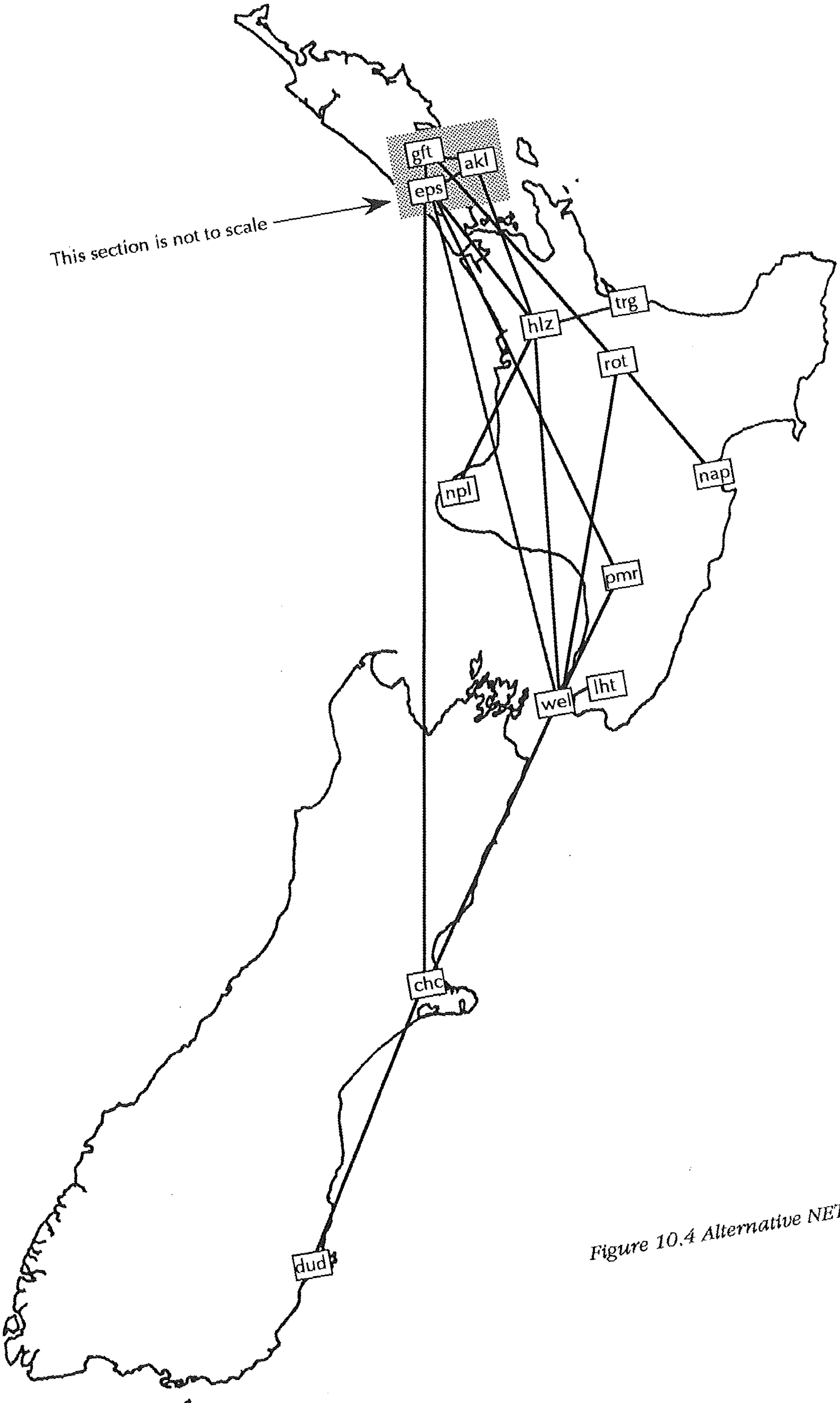

Figure 10.4 Alternative NET-X topology (geographical)

A COMPARISON OF NETWORK STRUCTURES

In order to quantify the two network structures, comparisons have been made on the basis of:

- Topological attributes;
- Average network delay;
- Average virtual circuit path length;
- Network capacity;
- Network cost components.

Attribute	Existing NET-X design	NDW Design
Topology		
Number of switches	16	13
Number of backbone links	22	17
Performance		
Average network delay (ms)	58.64	56.20
Average path length (hops)	1.22	1.12
Throughput (Kb/s @ 80ms)	143.23	170.77
Annual Cost		
Hardware	$ 387548.00	$ 353032.00
Backbone network	$ 374146.00	$ 287166.00
Access network	$ 570580.00	$ 607648.00
TOTAL ANNUAL COST	$ 1332274.00	$ 1247846.00

Table 10.b Comparison of network designs

The final design produced using NDW represents a structure that has three nodes more than the optimum[8], but three less than the existing design. The *Grafton* and *Auckland* nodes were not essential for decreasing cost, hence their position in the cost vs. dispersion curves of figure 10.2. In light of the traffic distribution results gained by the data analysis phase however, it was essential that extra nodes where deployed there.

[8] The optimum cost topology (using ADS for the access network) is illustrated in figure 10.8.

It is also important that the new structure fulfils the third design objective with the ability to absorb increased user traffic requirements. Figure 10.5 compares the traffic capacity of the two designs[9].

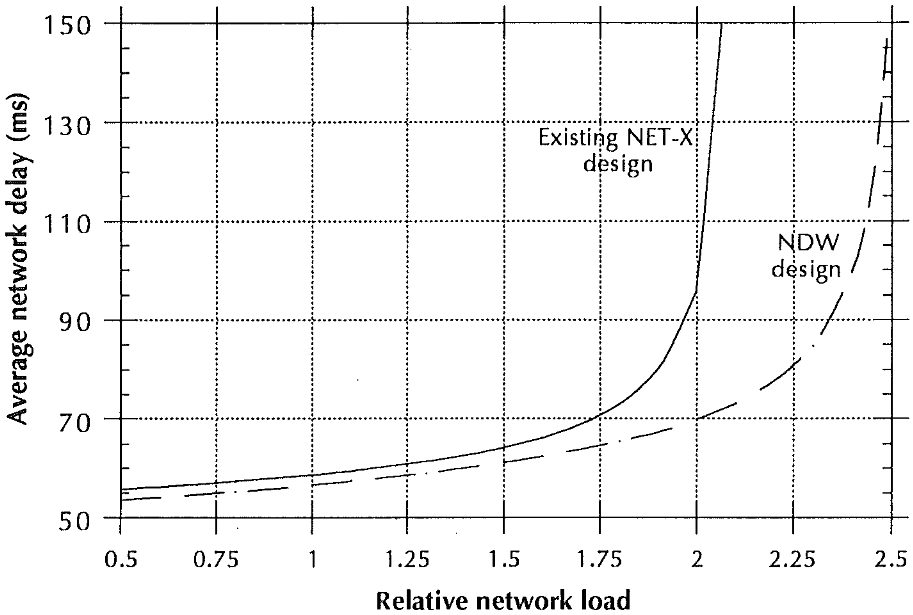

Figure 10.5 A comparison of network capacities

One can see from table 10.b and figure 10.6 that the new design represents a configuration that exhibits an 18% increase in throughput and saves 6.34% or $84,427.00 a year in running costs.

DEFICIENCIES IN THE EXISTING NET-X DESIGN

NDW has provided some insight into the cost-effectiveness of the current NET-X structure. The following deficiencies in the existing design have been highlighted.

❑ The current NET-X topology has too many nodes. The prime candidates for removal are (in order of priority): *Tamaki, Timaru* and *Invercargill.* The financial investment in these nodes does not compensate for the savings in access costs they achieve. For example, the Tamaki node serves only 8 subscribers and yet it represents an investment of $48,386.00. The removal of this node will not increase the access costs because its subscribers will be routed to one of the three remaining nodes in the Auckland area.

❑ The existing backbone structure is not 'in tune' with the current traffic distributions. The analysis presented in Chapter nine and subsequent exploration with NDW revealed the following shortcomings.

---

[9] Calculated using NDW's sweep function. Refer to page U-20 of Appendix III for more details.

There exist four severely underutilized links: the Epsom to Tamaki, Lower Hutt to Palmerston North, Dunedin to Invercargill and Napier to Rotorua links are utilized less than 2% during the peak hour and collectively cost $26,720 a year to maintain.

Fifty-five percent of the traffic exiting the Christchurch node is bound for hosts connected to the Epsom node. Terminating the Christchurch to Auckland link at Epsom as shown in figure 10.3 would improve traffic distributions[10].

Table 10.c illustrates the effects of moving this link.

Attribute	With Chc to Akl link	With Chc to Eps link
Average network delay (ms)	58.6	55.6
Average vc path length (hops)	1.2	1.1
Chc to Wel link flow (PPS)	16.0	6.0

Table 10.c Effects of relocating the Christchurch to Auckland link.

IMPACT OF DESIGN PARAMETERS

In addition to developing an alternative configuration for the NET-X network, NDW was also used to explore the impact of several design parameters on the overall network structure. To achieve the results summarised in figure 10.6 the following parameters where altered.

NODAL COST ALTERATION

Modern network systems provide the network engineer with a wide variety of network switch alternatives. Typically, the designer is faced with the option of deploying one large expensive node or several smaller low cost nodes.

To assess the effect of switch cost on the overall design, two options where considered:

- *Expensive nodes* — The use of CP9000 nodes configured with 9088 PSCs throughout the design. Each node has a processing capacity of 202 PPS, a maximum link fan-in of 512 ports, and an average price of $80,000.
- *Low cost nodes* — Limiting the design to CP9760 nodes configured with 9086 PSCs. Each node has a processing capacity of 202 PPS, a maximum link fan-in of 72 ports, and an average price of $38,000.

---

[10] A reduction of the transient traffic on the Christchurch to Wellington to Epsom links.

ACCESS NETWORK COST ALTERATION

Unfortunately there is only one link vendor in New Zealand so choices for the access links were limited. This monopoly is not the case overseas and network designers are presented with a much wider variety of link alternatives. To study the impact of access link costs on the final topology the following tariff structures were considered:

- *DDS access* — All access links utilize Telecom's DDS service[11].
- *ADS access* — Telecom's DDS is used for intercity links and the ADS for intracity links[12].

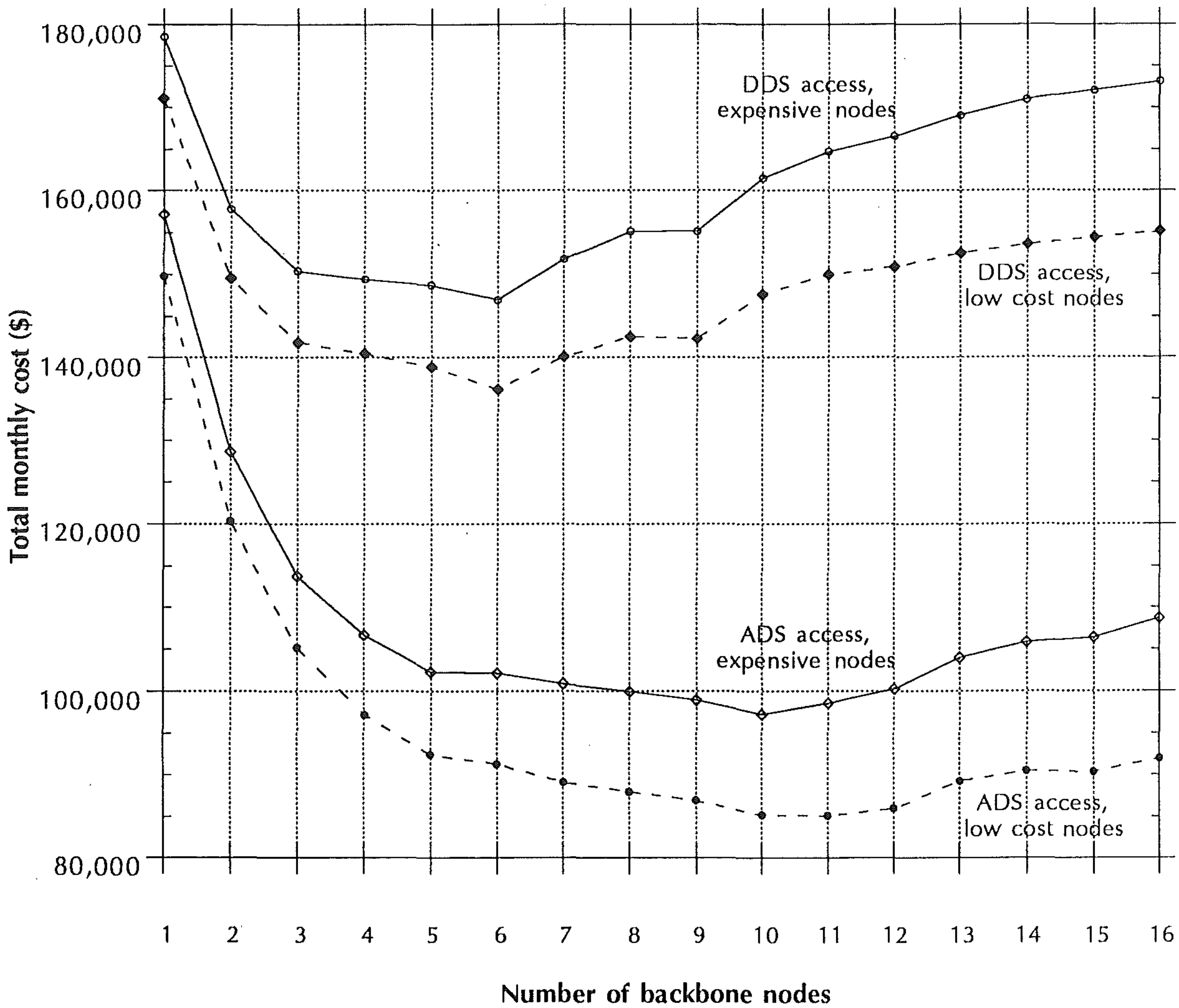

Figure 10.6 Effect of design parameters on the optimum number of backbone nodes

---

[11] Figure 10.1 details the cost vs. nodal dispersion curves for the DDS scenario.

[12] Figure 10.2 details the cost vs. nodal dispersion curves for the ADS scenario.

For each option, the optimum cost topology was designed to accommodate the required level of performance. Figures 10.7 and 10.8 show the topologies that emanated from the DDS access and the ADS access designs[13]. These topologies correspond to the cost minima in figure 10.6.

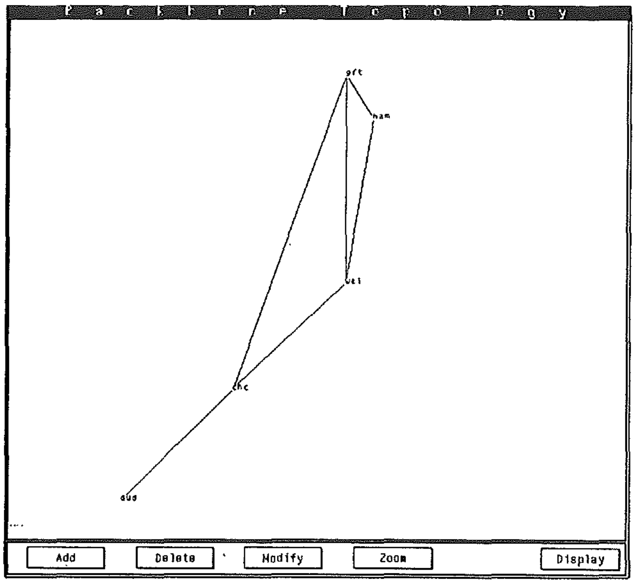

Figure 10.7 Minimal cost topology using DDS access links

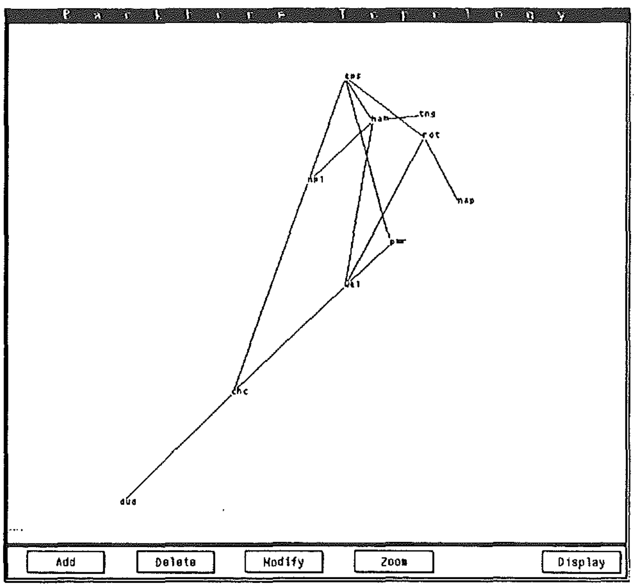

Figure 10.8 Minimal cost topology using ADS access links

13 Using CP9000 nodes.

DESIGN PARAMETER CONCLUSIONS

The graph in figure 10.6 illustrates the total annual network cost as a function of the number of network nodes for varying switch costs and access tariffs. From these results the following properties are observed:

❑ The access network tariff structure has a substantial bearing on the optimum number of nodes and the resulting topology. The optimum number of nodes is dependent on the rate of decrease in the access cost curve. This relationship is reflected in the curves of figures 10.1 and 10.2.

The DDS tariff has a high intracity rate compared to the ADS service[14], therefore the cost savings incurred by node placement are much less. This high intracity rate results in a decrease in the optimum number of switches (six compared with eleven). The minimal cost topologies represented by figures 10.7 and 10.8 bear out this conclusion.

❑ Given that subscriber numbers are sufficiently large, the switch cost has little impact on the optimum number of nodes.

The low cost node option resulted in a slight increase in the optimum number of nodes for the ADS design. This is illustrated by the lower two curves in figure 10.6. The decrease in nodal cost did not affect the DDS design.

These results reinforce the need for an integrated approach to network design. Such approaches will provide structures that are closer to the global cost minimum.

Based on these results it appears that the placement of the backbone nodes and the design of the subscriber access network form the most critical portion of the network design process.

CONCLUDING REMARKS

This chapter has illustrated the use of NDW in the redesign and planning stages of the NET-X packet switching network.

In addition to providing the designers with an improved structure, NDW has been used to study the impact of several design parameters on the optimum network topology.

Experimental results based on the design assumptions detailed in an earlier section of this chapter show that the optimum number and location of network nodes is critically dependent on the subscriber access network tariff structures. Because nodal costs are overshadowed by link costs the optimum number is less dependent on the hardware cost component.

---

[14] Resulting in the DDS access curve flattening off much sooner than the ADS curve.

The network designer is faced with a broad spectrum of design alternatives and a complex interdependence between them— which in itself highlights the importance of a structured design methodology and a powerful design tool.

CONCLUSIONS

This study was initiated with a clear goal in mind — to develop several computer-aided analysis and design tools for the subsequent evaluation and optimization of a nationwide X.25 packet-switching service.

During the course of this study advances were made in three primary areas: network design tools, network design methodologies, and finally the NET-X network. This final chapter provides a concluding look at the contributions made by this research.

NETWORK DESIGN TOOLS

The ever increasing complexity of modern network design mandates the use of sophisticated CAND tools. Having surveyed the field, it appears that the majority of contemporary design tools are deficient in one or more of the following areas— ease of use, graphics capabilities, effective feedback and the interaction with data collection software.

This thesis has presented Network Designers Workshop, a graphically orientated design tool that addresses these shortcomings. The main attractions of the NDW approach are: a high degree of man-machine interaction through extensive graphics facilities, ease of use, selective and useful feedback through the *design adviser* facility, and finally, the ability to directly import network management data.

The underlying strategy surrounding the implementation of NDW was to provide the network engineer with an *effective* design framework that augments the natural intelligence of the user rather than trying to totally replace it with an expert system.

NDW has utilized modified versions of published network design methods[1] with special emphasis being placed on providing the tool with a high degree of user-tool interaction. The practical consequences of the NDW approach mean the designer spends less time configuring the tool and tracking facts about the on-going design. With NDW they can 'home in' on the basic network structure faster

---

[1] These modifications are discussed in Chapter seven.

and spend more time evaluating alternatives and fine tuning the final design. This is one of the main innovative features of Network Designers Workshop.

The design tools developed during this study have been successfully applied to two nationwide networks[2] and have saved many man-hours in the analysis, reconfiguration, and cost optimization of these systems.

In addition, NDW was used to explore the relationship between the optimum cost network structure and several design variables.

FUTURE EXTENSIONS

The development of a design tool is an on-going process. NDW has been constructed with this in mind and as a result is very open-ended. This section discusses three possible extensions to NDW and indirectly treats some of its limitations.

The modular nature of NDW has ensured that future extensions may be carried out with a minimum of effort. Future enhancement may be focused on three areas:

- *Improved performance modelling* — NDW currently employs fairly rudimentary analytical techniques. NDW's performance prediction methods could be extended to use simulation techniques. The analytical methods would still form the essential front-end for the fast prototyping of designs, with the simulation methods being used for the verification of final designs.
- *Applicability to other network designs* — NDW has been developed for the design of distributed packet relay networks, although the tool could easily be adapted for many other network applications. Methods could be added for the design of cell relay networks, circuit switched networks, centralized multi-drop topologies and so on.
- *Improved design quality for large scale networks* — The integrated design method adopted by NDW can be classified as a greedy algorithm. NDW's synthesis method makes a decision on which node to add or which link to add, upgrade or remove on the sole basis of the maximized improvement at that stage of the optimization. This step by step approach has a major drawback— the criterion for choosing the next configuration n_{i+1}* is based entirely on the network configuration n_i* at that step. No allowance is made for the impact of the design action at a later stage of the optimization process. Furthermore no facility exists for remedying an inappropriate action at step i. Whilst greedy methods perform well on small to medium sized networks, more sophisticated design methods may be required for the design of large scale networks. Fortunately several techniques exist including lookahead methods [31] and

---

[2] Refer to Chapter ten and Appendix VI.

genetic algorithms [100]. These methods consider a much broader range of options at each step[3] with the result of providing better long-term solutions.

THE NET-X NETWORK

NET-X is an established and growing network which provides a packet switched service to several hundred subscribers around New Zealand. The continued success of NET-X depends primarily on the maintenance of a cost-effective configuration that has the ability to expand and adapt to changing user demands.

The computer based design tools developed during this study have proved invaluable for the re-evaluation of the current NET-X design. The tools have provided the NET-X engineers with a considerable amount of insight into the operational state of their network and have collectively been used in a network improvement study.

NDW has been used to highlight and rectify design faults in the current NET-X configuration. A new topology has been proposed[4] which achieves an 18% increase in traffic carrying capacity with an annual saving of $84,427.00.

It is pleasing to note that several alterations to the NET-X topology being put into practice coincide with the suggestions made by this study. Chief among these is the removal of the Tamaki, Timaru, and Invercargill nodes and their respective backbone links. Additionally, the remaining links connecting the Auckland, Grafton, and Epsom cluster have been reconfigured to mirror the configuration shown in figure 10.3.

DESIGN METHODS

This study has illustrated the need to adopt integrated design methods to capture the important interdependence that exists between node placement and backbone design. It appears that the use of integrated design techniques such as the one proposed in Chapter six offer the potential for far greater cost savings than the isolated use of node placement, access network, and backbone design methods.

Discussions in Chapter three have pointed out that the use of heuristic algorithms are still the only practical approach to solving the network design problem. Chapter seven highlighted several shortcomings of Chou's generalized cut-saturation heuristic and has proposed several improvements to it.

Because of their inherent nature, design heuristics do not always provide the best solutions. At the present time, the only way to ensure an effective design is through the successful integration of the practitioner and a sophisticated design

---

[3] With the added cost to speed and complexity.

[4] The optimized topologies are detailed in figures 10.3 and 10.4

tool. Because of the involvement of human judgment, network design— at least for the near future— will remain a combination of both art and science.

The latter sections of Chapter ten have highlighted the impact of several design parameters on the optimum topology. Results[5] based on the NET-X environment show that the optimum topology is critically dependent on the access link tariff structure and less so on the backbone switch cost. These results suggest that the placement of backbone nodes and the design of the access network form the most critical part of the network design process.

FINAL COMMENT

The operators of todays data communication networks enjoy the advantages of a wide variety of system options. They also face the challenge of network planning and analysis to ensure a cost-efficient system which supplies its subscribers with an adequate class of service. These challenges can be met only by the use of sophisticated Computer-Aided Network Design tools.

The graphical approach adopted by NDW provides a pragmatic and highly desirable environment in which the user can actively participate in the design process. It is anticipated that the adoption of interactive graphics for network design tools will grow rapidly as an increasing number of software developers become aware of its benefits and the availability of graphics workstations continue to flourish.

Network Designers Workshop could serve as a prototype for the continued development of sophisticated design tools.

[5] As illustrated in Figures 10.6, 10.7 and 10.8.

appendix I

TELECOM LEASED CIRCUIT TARIFFS

This appendix details the leased circuit tariffs used in this study. Prices and services are effective as of 21 April 1991 and exclude the Government Goods and Services Tax (12.5%).

Three elements are used to determine the total charge for the three services below namely:

- A once only installation charge per circuit end;
- A monthly access charge per circuit end;
- A monthly transmission charge, based on Telecom's tariff-zoning scheme.

TARIFF ZONING SCHEME

Telecom base their distance dependent transmission charge on a zoning scheme. New Zealand has been divided into 17 zones with the monthly transmission charge being dependent on the number of zones the link extends over. Six zone dependent rates exist (CITY, A, B, C, D, E); the lowest CITY, for intracity links, next A for intrazone links and the highest E for links between the most distant zones.

Table I.a below details the charging steps for the 17 zones.

	WR	AK	HN	RO	NU	GS	NA	WG	PM	WN	MS	NN	CH	GM	TU	DN	IN
WR	A	B	C	D	D	D	D	D	D	D	D	E	E	E	E	E	E
AK	B	A	B	C	C	D	D	D	D	D	D	D	E	E	E	E	E
HN	C	B	A	B	B	C	C	C	C	D	D	D	E	E	E	E	E
RO	D	C	B	A	C	B	B	B	B	C	C	D	D	D	E	E	E
NU	D	C	B	C	A	C	C	B	B	C	C	C	D	D	D	E	E
GS	D	D	C	B	C	A	B	C	C	C	C	D	D	D	E	E	E
NA	D	D	C	B	C	B	A	B	B	B	B	C	D	D	D	E	E
WG	D	D	C	B	B	C	B	A	A	B	B	C	D	D	D	D	E
PM	D	D	C	B	B	C	B	A	A	B	B	C	D	D	D	D	E
WN	D	D	D	C	C	C	B	B	B	A	A	B	C	C	D	D	E
MS	D	D	D	C	C	C	B	B	B	A	A	B	C	C	D	D	E
NN	E	D	D	D	C	D	C	C	C	B	B	A	B	B	C	C	D
CH	E	E	E	D	D	D	D	D	D	C	C	B	A	B	B	C	D
GM	E	E	E	D	D	D	D	D	D	C	C	B	B	A	B	B	C
TU	E	E	E	E	D	E	D	D	D	D	D	C	B	B	A	B	C
DN	E	E	E	E	E	E	E	D	D	D	D	C	C	B	B	A	B
IN	E	E	E	E	E	E	E	E	E	E	E	D	D	C	C	B	A

Table I.a Leased circuit transmission charging steps

ANALOGUE DATA SERVICE

Installation charges (per end)

• A1/A2	$420.00
• A3	$500.00

Monthly access charges (per end)

	(to 5km)	Distance surcharge (per km)+
• A1	$31.82	$8.50
• A2	$63.64	$12.50
• A3	$63.64	$12.50

Monthly transmission charge

	A	B	C	D	E
A1	$ 550.00	$ 800.00	$ 1120.00	$ 1660.00	$2150.00
A2	$ 550.00	$ 800.00	$ 1120.00	$ 1660.00	$ 2150.00
A3	$ 600.00	$ 800.00	$ 1230.00	$ 1830.00	$ 2370.00

DIGITAL DATA SERVICE

Installation charges (per end)

• 2k4, 4k8, and 9k6	$280.00 per NTU*
• 19k2, 48k, 64k, and 128k	$1680.00 per NTU

Monthly access charges (per end)

• 2k4 bps	$160.00
• 4k8 bps	$170.00
• 9k6 bps	$190.00
• 19k2 bps	$300.00
• 48k bps	$330.00
• 64k bps	$650.00
• 128k bps	$800.00

Monthly transmission charge

	CITY	A	B	C	D	E
2k4 bps	$ 30.00	$ 30.00	$ 50.00	$ 90.00	$ 140.00	$ 160.00
4k8 bps	$ 45.00	$ 50.00	$ 90.00	$ 150.00	$ 270.00	$ 290.00
9k6 bps	$ 55.00	$ 70.00	$ 150.00	$ 250.00	$ 400.00	$ 530.00
19k2 bps	$ 240.00	$ 310.00	$ 510.00	$ 835.00	$ 1200.00	$ 1330.00
48k bps	$ 240.00	$ 310.00	$ 610.00	$ 1140.00	$ 1820.00	$ 2260.00
64k bps	$ 240.00	$ 310.00	$ 610.00	$ 1140.00	$ 1820.00	$ 2260.00
128k bps	$ 480.00	$ 620.00	$ 1220.00	$ 2280.00	$ 3640.00	$ 4520.00

+ Applies when the access link is longer than 5 km.

* NTU = Network Terminating Unit, provided by Telecom as interface to subscriber's DTE.

MEGALINK SERVICE

Installation charges (per end)

- All speeds (1 × 64K to 2MB) $7500.00

Monthly access charges (per end)

- All speeds (1 × 64K to 2MB) $2000.00

Monthly transmission charge

No of 64K circuits	CITY	A	B	C	D	E
1	$ 240	$ 310	$ 610	$ 1140	$ 1820	$ 2260
2	$ 480	$ 620	$ 1220	$ 2280	$ 3640	$ 4520
3	$ 720	$ 930	$ 1830	$ 3420	$ 5460	$ 6780
4	$ 960	$ 1240	$ 2440	$ 4560	$ 7280	$ 9040
5	$ 1200	$ 1550	$ 3050	$ 5700	$ 9100	$ 11300
6	$ 1440	$ 1860	$ 3660	$ 6840	$ 10920	$ 13560
7	$ 1680	$ 2170	$ 4270	$ 7980	$ 12740	$ 15820
8	$ 1920	$ 2480	$ 4880	$ 9120	$ 14560	$ 18080
9	$ 2160	$ 2790	$ 5490	$ 10260	$ 16380	$ 20340
10	$ 2400	$ 3100	$ 6100	$ 11400	$ 18200	$ 22600
11	$ 2640	$ 3410	$ 6710	$ 12540	$ 20020	$ 24860
12	$ 2880	$ 3720	$ 7320	$ 13680	$ 21840	$ 27120
13	$ 3120	$ 4030	$ 7930	$ 14820	$ 23660	$ 29380
14	$ 3360	$ 4340	$ 8540	$ 15960	$ 25480	$ 31640
15	$ 3600	$ 4650	$ 9150	$ 17100	$ 27300	$ 33900
16	$ 3840	$ 4960	$ 9760	$ 18240	$ 29120	$ 36160
17	$ 4080	$ 5270	$ 10370	$ 19380	$ 30940	$ 38420
18	$ 4320	$ 5580	$ 10980	$ 20520	$ 32760	$ 40680
19	$ 4560	$ 5890	$ 11590	$ 21660	$ 34580	$ 42940
20	$ 4800	$ 6200	$ 12200	$ 22800	$ 36400	$ 45200
21	$ 5040	$ 6510	$ 12810	$ 23940	$ 38220	$ 47460
22	$ 5280	$ 6820	$ 13420	$ 25080	$ 40040	$ 49720
23	$ 5520	$ 7130	$ 14030	$ 26220	$ 41860	$ 51980
24	$ 5760	$ 7440	$ 14640	$ 27360	$ 43680	$ 54240
25	$ 6000	$ 7750	$ 15250	$ 28500	$ 45500	$ 56500
26	$ 6240	$ 8060	$ 15860	$ 29640	$ 47320	$ 58760
27	$ 6480	$ 8370	$ 16470	$ 30780	$ 49140	$ 61020
28	$ 6720	$ 8680	$ 17080	$ 31920	$ 50960	$ 63280
29	$ 6960	$ 8990	$ 17690	$ 33060	$ 52780	$ 65540
30	$ 7200	$ 9300	$ 18300	$ 34200	$ 54600	$ 67800
2Mb	$ 3600	$ 6510	$ 14640	$ 27360	$ 43680	$ 54240

appendix II

HARDWARE COST SCHEDULE

The following sections detail the hardware components used in this study. Prices are effective as of 21st of April 1991 and exclude the Government Goods and Services Tax (12.5%).

NODE CHASSIS

The following prices correspond to the base price[1] of the backbone nodes used in this thesis.

❐ **CP 9000 NPX (Network Packet Exchange)**	$22,846.00
• One Rack	
• One Communications Chassis (12 Slots)	
• Redundant Power Supplies	
• Up to 512 RS-232 Ports	
• Modem rack	$1960.00
	$24,806.00

❐ **CP 9760 NPX (Network Packet Exchange)**	$17,925.00
• Office Cabinet	
• Single Chassis (12 Slots)	
• Full Power Supply Redundancy	
• Up to 72 RS-232 Ports	
• Modem rack	$1960.00
	$19,885.00

❐ **CP 9724 NPX (Network Packet Exchange)**	$4501.00
• Pedestal Cabinet	
• Up to 24 Ports	
• Modem rack	$1960.00
	$ 6461.00

[1]Refer to equation 4.8 in Chapter four.

NODE PROCESSING MODULES

Hughes 9088 packet switching clusters were chosen to configure the network nodes in this study. The 9088 price corresponds to the module cost component of equation 4.9 in Chapter four.

❒ **9088 Twelve Port High/Medium Speed Cluster** **$41,925.00**

- One PM and Three LIMs
- 12 lines to 19.2 kbps

  or
- 6 lines at 56/64 Kbps

MODEMS

Two Codex 2345 modems were assumed when pricing an Analogue Data Service link. This cost corresponds to the $C_{interface}$ component of equation 4.6 discussed in Chapter four.

❒ **Codex 2345 Modem** **$1000.00**

- 9600bps

a p p e n d i x I I I

A USERS GUIDE TO NDW

This appendix contains the standard users guide to Network Designers Workshop.

This user guide forms the primary document detailing the use of the program.

NETWORK DESIGNERS WORKSHOP NDW

A USERS GUIDE

CONTENTS

INTRODUCTION

Network Designers Workshop (NDW) is a software package designed for the evaluation and synthesis of computer communication networks. NDW evolved from an industry requirement for improved tools to assist in the design and management of commercial data networks.

NDW was designed and developed during 1990-1991 by Clinton Logan (University of Canterbury, Christchurch, New Zealand).

This guide is divided into three main sections:

- INTRODUCTION (the section you are reading now): Provides a general overview of NDW.
- SYSTEM OPERATION: Details the facilities provided by NDW and describes how to use them.
- COMMAND REFERENCE: Provides a complete description of NDW's commands.

SYSTEM OVERVIEW

NDW has been specifically designed to provide the following:

- Ease of use;
- A responsive visually orientated environment;
- Intelligent feedback;
- Interaction with network management software.

NDW draws on the desktop metaphor for its primary means of user interaction. NDW is totally menu driven by way of a central console window. Windows can be created, moved, and destroyed, each providing informative feedback on a particular network aspect (i.e. costs, delays, utilizations etc.). Figure 1.1 details a 'snapshot' of a NDW desktop during a typical design session.

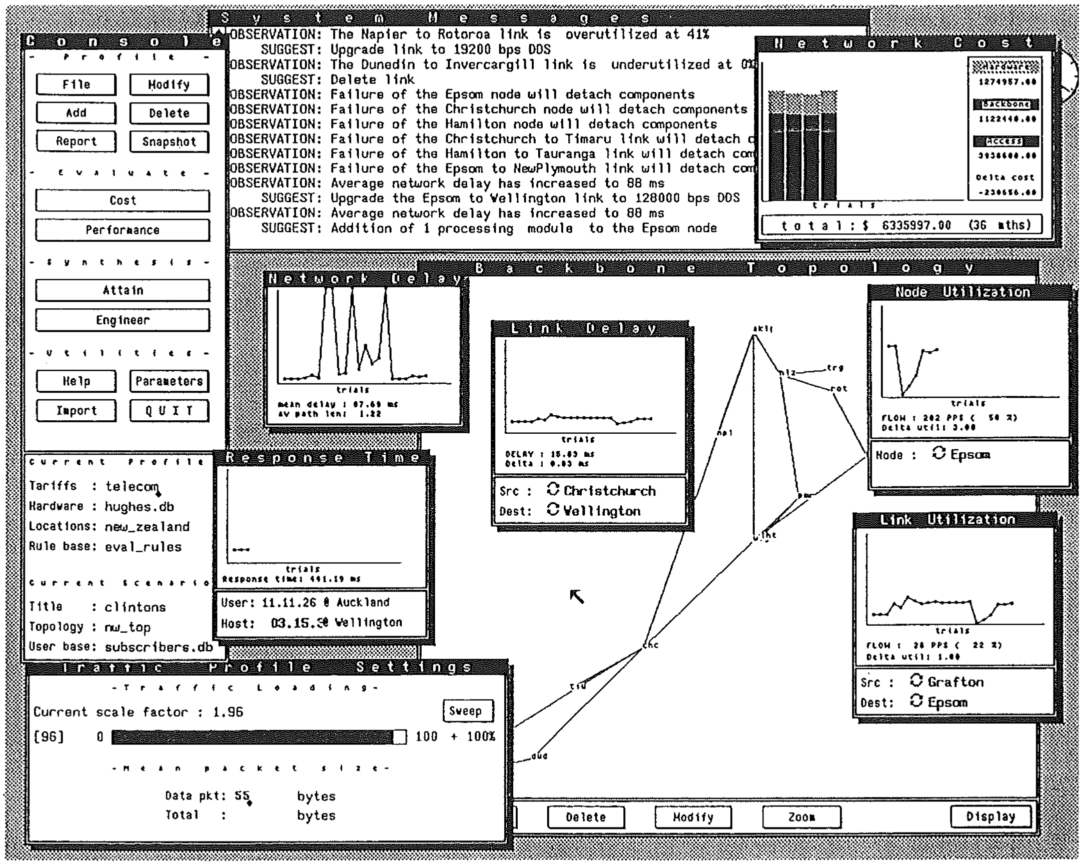

Figure 1.1 NDW desktop

Illustrated in figure 1.1 are the four main window types maintained by NDW.

• CONSOLE: The console window consists of two panels. First, the menu panel — all user commands are issued using the 14 popup menus on this panel; Second, the status panel detailing the names of all currently loaded files.

• TOPOLOGY : The topology window displays the current network topology and several modification menus.

• SYSTEM MESSAGES: All system messages and suggestions are relayed to the user through this window.

• MONITORS: There exists several types of monitor windows (several are shown in figure 1.1). These windows enable the user to monitor network attributes (e.g. node utilization, network cost, response time etc.) throughout the design process.

Special features

NDW possesses several unique properties, including:

• **Ease of use**

All user interaction is by way of a graphical user interface. All commands are logically structured and issued using popup menus. NDW also provides comprehensive error checking and on-line help facilities.

• **High levels of user-tool interaction**

NDW employs high resolution graphics to instantly display the impact on cost and performance as the user investigates "what if" scenarios. The vast majority of contemporary tools provide feedback in the form of generated reports thus restricting the practitioner to an almost batch mode style of operation. With NDW, the designer can immediately observe the effects of altering network variables; for example, "What happens to the average virtual circuit path length if I delete this link?," or "What are the consequences of shifting this host?, increasing the traffic?, altering this node?, adding this user?" etc.

• **Intelligent feedback**

NDW draws on AI techniques to blend some of the designer's skill with the tool. NDW uses a rule based approach to automatically verify and evaluate the designer's actions. Rules can be constructed (via an editor) to ensure the on-going network design fulfils certain cost, performance, and reliability criteria. If the design strays beyond these constraints NDW will notify the user and suggest possible solutions.

• **Interaction with network management software**

NDW allows the direct importation of operational data from the network management facilities existing on most modern systems. This data is imported using external filters.

System architecture

NDW is a modular tool. The designer can use one particular module to perform a specific task (e.g. cost a link) or coordinate the use of all modules for a complete network design. NDW's architecture is illustrated in figure 1.2.

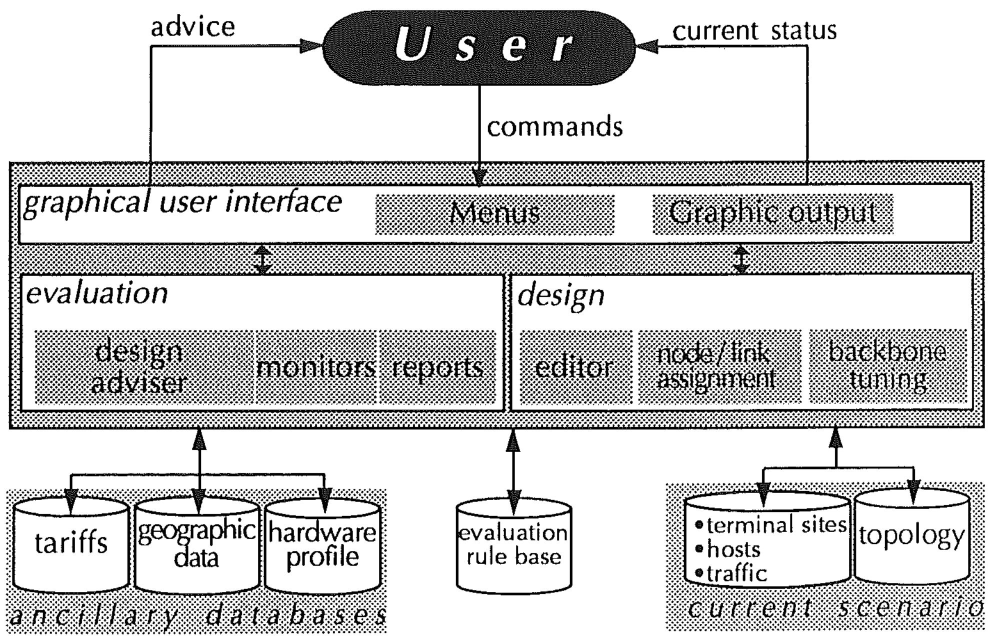

Figure 1.2 NDW system architecture

Program requirements

NDW is designed to operate on a SUN™ Workstation (preferably a Sparcstation 1 or 2) under Suntools or Open Windows. NDW will work with a colour or monochrome monitor, however colour is necessary to take advantage of the topology coding features.

NDW design sessions involve many thousands of floating point operations making a floating point accelerator chip mandatory for ensuring acceptable responsiveness.

A minimum of 1.25 Mb virtual memory and 500Kb of disk space are required by NDW.

SYSTEM OPERATION

This section provides an overview of the facilities provided by NDW and describes how to use them. The topics are organized into the category of tasks that you would normally perform during a design session.

This guide assumes the user is familiar with the GUI objects (e.g. popup menus, choice items, sliders etc) offered under the Suntools or Open Windows environment.

SPECIFYING NETWORK DETAILS

NDW's facilities for allowing the user to define or alter information for a new or existing network are described within this section. The setting up of a network profile is the first step in any design process. NDW stores network information as a series of files. Several different groups of these files can then be saved and reloaded to form network scenarios. The user can use these scenarios to keep track of different design alternatives.

The consequences of any network alterations are immediately displayed within the currently active monitor windows. If the user has an evaluation rulebase loaded, the network will be evaluated accordingly (see the network evaluation section for more details).

Nodes

The following actions can be performed on the nodes within a network:

• **Add a Node** — A node can be added using the *Add* menu in either the console or topology windows. Nodal information can then be entered using the dialog box shown in figure 3.5.

The node's location and hardware type can be selected using the corresponding popup choice items. The node is then plotted within the topology window.

• **Delete a Node** — A node can be deleted from the current topology using the *Delete* menu in either the console or topology windows. The node is selected using the popup choice item in the resulting dialog box.

NOTE: All backbone links terminating at that node will also be deleted.

• **Modify a Node** — Node details can be edited using the *Modify* menu in either the console or topology windows. Once the node has been selected from the popup choice item, the user can alter any of the node's details.

Backbone links

The following actions can be performed on network backbone links:

• **Add a Link** — A link can be added to the backbone using the *Add* menu in either the console or topology windows. Link details can then be entered using the dialog box in figure 3.6.

The link's source, destination, speed, and type can be selected using the corresponding popup choice items. The new link is then displayed within the topology window.

• **Delete a Link** — A link can be deleted from the current topology using the *Delete* menu in either the console or topology windows. The link is selected using the popup choice in the dialog box shown in figure 2.1.

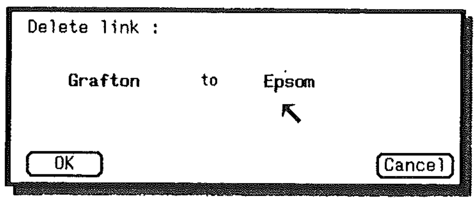

Figure 2.1 Link Choice Dialog

• **Modify a Link** — Link details can be altered using the *Modify* menu in either the console or topology windows. Once the link has been selected, the user can then alter any of the link's details.

Host specification

NDW maintains information on all hosts connected to the network. The following actions can be performed on the host data base:

• **Add a Host** — A new host can be added to the network using the *Add* menu. The user can then enter the host's details.

From this point on, the new host will appear in all relevant popup choices.

• **Delete a Host** — A host can be deleted using the *Delete* menu. The user is then forced to update all subscribers that were previously homed to that host.

• **Modify a Host** — Host details can be altered using the *Modify* menu in the console window. The user is then prompted for the host's name. Once found, the host's details can be altered.

Terminal centre

The subscriber database contains information on all terminal centres connected to the network. Figure 3.1 (in the command reference section) details the data entry dialog presented to the user whenever a new terminal centre is created, or the details of one changed.

The following actions can be performed on the currently loaded terminal centres:

• **Add a Terminal centre** — A new centre can be added to the network using the *Add* menu in the console window. Details are then entered using the dialog box in figure 3.1.

• **Delete a Terminal centre** — Any existing centre can be deleted from the network using the *Delete* menu in the console window.

• **Modify a Terminal centre** — Terminal centre details can be altered using the *Modify* menu in the console window. The user is then prompted for the centre's name. Once found, details can be altered and re-saved.

Hardware profiles

NDW maintains a file that contains profiles of network hardware objects (e.g. nodes, PADs, FEPs etc.) and their corresponding attributes (e.g. cost, maximum throughput, maximum number of ports etc.). By entering these profiles, NDW effectively builds an inventory of the equipment that can be used in a specific design. Hardware profiles can be added, deleted, or edited as follows:

• **Add a hardware description** — A new hardware description can be added to the network scenario using the *Add* menu in the console window. Details are then entered via the dialog box shown in figure 3.4

• **Delete a hardware description** — Any existing hardware description can be deleted from the network using the *Delete* menu in the console window. The user will then be forced to alter any existing network components that are based on that hardware.

• **Edit an existing hardware description** — Hardware details can be altered using the *Modify* menu in the console window. The particular hardware profile is selected from the hierarchical *hardware profile* command within the *Modify* menu.

Protocol descriptions

NDW stores definitions for communication protocols. This enables the program to model various protocol-specific transmission overheads. Protocol definitions can be added, deleted, and edited as follows:

• **Add a protocol definition** — A new protocol definition can be added using the *Add* menu in the console window. Details are then entered by way of the dialog box shown in figure 3.7.

• **Delete a protocol definition** — Any existing definition can be deleted from the network using the *Delete* menu in the console window. The user will then be forced to alter any existing network components that use that protocol.

• **Modify an existing protocol definition** — A protocol definition can be altered using the *Modify* menu in the console window. The protocol is then selected from the hierarchical *protocol definition* command within the *Modify* menu in a similar fashion to modifying a hardware profile.

Traffic profile

Central to the design process is the specification of the network traffic flows. The current point-to-point traffic flows can be defined using the following two mechanisms:

• Each subscriber record (see figures 3.1 and 3.2) contains two fields for entering the total traffic transmitted between the subscriber and its destination host(s). Any host to host file transfers can be specified using the host specification mechanisms discussed above.

• The current (relative) load and packet size statistics can be specified using the *traffic profile* command within the *Parameters* menu (see figure 3.24).

NETWORK EVALUATION

NDW is primarily a tool for the evaluation of computer networks. During a typical design session the user will alter network variables (e.g. shift a host, increase the traffic, delete a link etc.) and observe the effect of doing so. NDW has several mechanisms that facilitate this process of "what if" exploration.

Report generation

At any stage during the design process the user can use NDW's reporting module to generate comprehensive reports. These reports are written to standard text files in a format that is both human readable and can be directly imported by microcomputer based graphing packages.
The command reference section contains detailed descriptions of these reports.

System monitors

The provision of a responsive visual environment is central to NDW's design philosophy. NDW enables the designer to monitor various network component attributes using monitor windows. The *Cost* and *Performance* menus provide the designer with the ability to monitor the following network attributes: total network cost, access and backbone link costs, node costs, user response times, link delays and finally link utilizations. A complete description of NDW's monitors can be found in the command reference section.

Design adviser

During a design session hundreds of variables will constantly undergo change. NDW's design adviser provides a powerful yet flexible mechanism for relieving the user of some of the design burden. The design adviser uses a rule based approach to automatically verify and evaluate the designer's actions. Evaluation rules can be constructed using the editor in figure 2.2 to ensure the on-going design fulfils certain cost, performance and reliability criteria. Should the design stray outside the constraints set by those rules, NDW will notify the user through the system messages window shown in figure 2.3 and suggest the "best" solution to the problem.

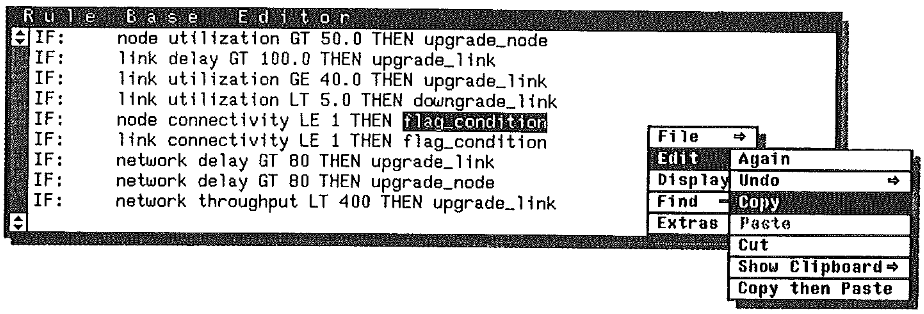

Figure 2.2 Rule base Editor

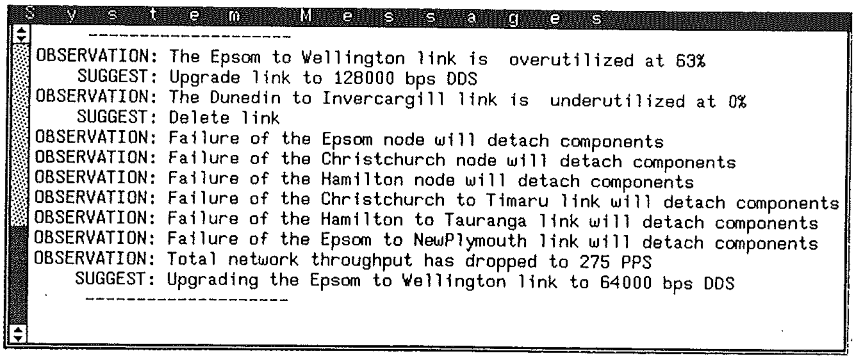

Figure 2.3 System Messages window

NDW's evaluation rules are constructed from a predefined group of primitives and must conform to the following syntax:

<rule> ::= (IF <antecedent> THEN <consequent> [ELSE <consequent >])
<antecedent> ::= <object> <attribute> <value>
<value> ::= <comparator><number>
<comparator> := LE / EQ / GT / GE / NE / LT
<object>::= node / link / network / subscriber
<attribute>::= utilization / throughput / delay / connectivity /cost
<consequent> := upgrade_node / upgrade_link / downgrade_node / downgrade_link / upgrade_component / downgrade_component / flag_condition

If the designer wants NDW to "keep an eye on" the network throughput, and to suggest the most cost-effective link to upgrade (or even add) should it fall below 200 Kb/s, then the following rule would be added to the rulebase:

IF: network throughput LT 200 THEN upgrade_link

Alternatively, should the user wish to know if any intra-nodal delays exceed 50ms, without wanting a solution:

IF: node delay GT 50 THEN flag_condition

If the user then decides that a 45ms delay should be considered excessive, the rule can be edited to:

IF: node delay GT 45 THEN flag_condition

Or removed altogether if the user doesn't want NDW to monitor nodal delays.
The table below details the rulebase primitives available to the designer:

<objects>	*<attributes>*	*<consequents>*
node	utilization	upgrade_node
link	throughput	upgrade_link
network	delay	downgrade_node
subscriber	connectivity	downgrade_link
	cost	upgrade_component
		downgrade_component
		flag_condition

Some example scenarios

NDW's responsive graphical displays greatly facilitate the answering of a multitude of "what if" questions. The impact of any alteration is immediately displayed within the currently active monitor windows. Additionally, NDW's design adviser will use a loaded rulebase to evaluate the network. A few examples are provided below:

- **"What will be the impact of shifting the HP3000 host from its present location in Epsom to a new location in Wellington?"** — To shift the HP3000 host, the user selects the *Modify host* command and enters the host's name. The host's location can be altered using the popup choice. As soon as the user confirms the change, NDW will automatically reassign the host to the nearest network node, calculate the change in traffic flows and display the results.
- **"Will the addition of these new terminal centres in Christchurch have any serious effect on the current network?"** — The user can add the new centres using the *Add terminal centre* command and specify their corresponding details. NDW will immediately calculate the effect of adding each subscriber and display the results.
- **"What is the traffic handling capacity of the present network?" and "What will be the first component to saturate?"** — The traffic sweep function can be used to select a range of traffic load values to test . If the *saturated component* halting condition is selected , the global traffic levels will be automatically scaled until a component saturates (i.e. experiences infinite delays). The component's details are then displayed within the system messages window.

- **"What will be the effect of deleting the Invercargill node?"** — The user can delete the Invercargill node using the *Delete node* command. NDW will automatically re-assign all subscribers to the nearest remaining node. All backbone links terminating at the Invercargill node will also be deleted. The effects on the network's cost and performance are recalculated and displayed.

NETWORK SYNTHESIS

As an alternative to evaluating an already existing network, a user may wish to design a completely new network. In this case, the user would have to define the subscriber base (using the subscriber specification procedures discussed in the previous section). NDW can then be used to design a cost-effective network structure using the facilities detailed below. The network's cost , performance and reliability attributes can then be analysed using the evaluation facilities provided by NDW.

Node placement

Given an existing subscriber profile, NDW can determine the "best" location for installing a new node. The user can issue the *node placement* command, by way of the *Attain* menu, to list all possible node locations and their corresponding cost savings. The designer can then follow NDW's recommendations, or decide to place a node in some other location. This facility can be used to add a new node to an already existing topology, or to determine the optimum node locations for a new network.

Link placement

Given the current nodal locations and the traffic requirements between them, the *Attain link placement* command can be used to find possible ways of interconnecting them. NDW will list possible links, along with their traffic requirement, cost, and a preference factor. This factor ranges from 1 (highly recommended) to 0.

Some example scenarios

- **"This network configuration is too expensive, what would be the most effective way of decreasing its cost?"** — If the *Attain cost decrease* command is issued, NDW will determine the most expensive and underutilized network component and suggest a downgrade option. NDW will ensure the downgrade option still satisfies the performance constraints set by the user.
- **"The current network throughput is too low, what would be the most cost effective way of increasing it?"** — Once the *Attain throughput increase* command has been issued, NDW determines the "best" network component to upgrade, resulting in an overall increase in network throughput.
- **"Given the current network configuration how can I reduce the mean end-to-end delay?"** — On issuing the *Attain Delay decrease* command, NDW highlights the components that are contributing the most to the mean network delay. NDW will list the most cost-effective link and node to upgrade in order to decrease this delay.

NETWORK TUNING

The topology tuning methods built into NDW attempt to improve on existing structures by finding a minimal cost layout that satisfies specified performance constraints. NDW uses a modified version of the *Generalized Cut-Saturation* heuristic for tuning backbone structures..

Backbone tuning

The user is prompted to enter the backbone tuning criteria using the dialog box shown in figure 3.18. Once the *Optimize* button is pressed, NDW will perform the heuristic design steps until one of the specified halting conditions is met.

COMMAND REFERENCE

NDW presents the user with 89 different commands logically contained within 19 popup menus. This section describes each menu command in detail.

FILE MENU

The *File* menu contains commands for opening, creating and saving NDW's network files.

New

- **Subscribers** — Creates a new list of subscribers.
- **Topology** — Deletes the current topology.
- **Network scenario** — Creates a new scenario.
- **Hardware profile** — Erases all currently loaded hardware descriptions. A new set of hardware profiles can now be created.
- **Locations** — Creates a new geographic location file.
- **Line Tariffs** — Creates a new line tariff file.
- **Evaluation rules** — Invokes the rulebase editor, enabling the user to create a new set of evaluation rules (see figure 2.2).

Load / Save

- **Subscribers** — Prompts the user for the name of a subscriber file to load/save.
- **Topology** — Displays a dialog box prompting the user for the name of a topology file to load/save.
- **Network scenario** — Prompts the user for the name of a scenario file to load/save.
- **Hardware profile** — Prompts the user for the name of a hardware file to load/save.
- **Locations** — Prompts the user for the name of a geographical location file to load/save.
- **Line Tariffs** — Prompts the user for the name of a line tariff file to load/save.
- **Evaluation rules** — Prompts the user for the name of an evaluation rulebase to load/save.

ADD / DELETE MENU

The commands in the *Add/Delete* menus are used to add and remove network components.

Terminal centre

- **ADD Terminal centre** — The user is prompted for the new centre's details using the dialog shown in figure 3.1.

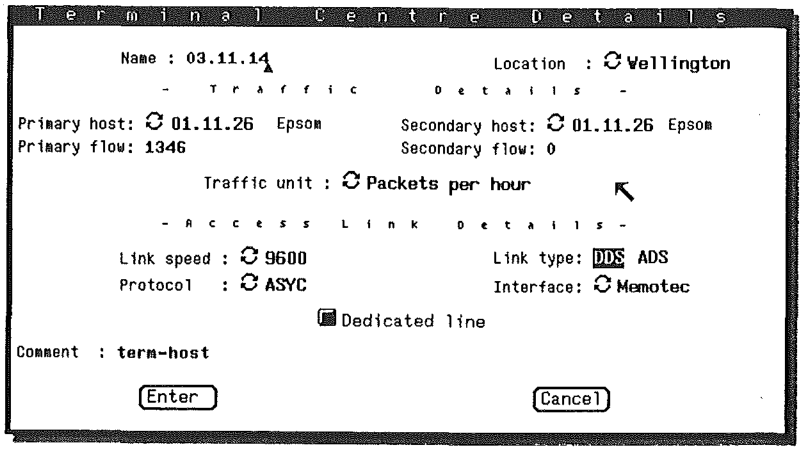

Figure 3.1 Terminal centre detail entry dialog

The following details must be specified:
NAME: A name or label identifying the centre.
LOCATION: The centre's location (selected from the popup list).
PRIMARY HOST: The centre's primary host (selected from the popup list).
SECONDARY HOST: The centre's secondary host (selected from the popup list).
PRIMARY FLOW: The traffic flowing between the terminal centre and its primary host.

SECONDARY FLOW: The traffic flowing between the terminal centre and its secondary host.

TRAFFIC UNIT: The traffic flow unit (selected from the popup list).

LINK SPEED: The access link speed (selected from the popup list).

PROTOCOL: The access link protocol (selected from the popup list).

LINK TYPE: The access link type (selected from the popup list).

INTERFACE: Any access equipment (e.g. PAD, FEP etc.).

DEDICATED LINE: Select the radio button if the terminal centre has a leased access line.

COMMENT: Any comment pertaining to this terminal centre.

- **DELETE Terminal centre** — The user is prompted for a terminal centre name. The centre is then deleted from the current working network scenario (not from the database). If the subscriber database is re-saved to disk, the centre will be permanently deleted.

Host

- **ADD host** — New host details can be entered using the dialog in figure 3.2.

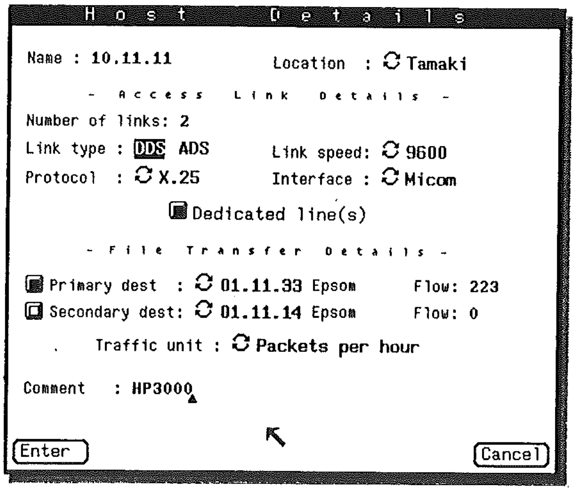

Figure 3.2 Host detail entry Dialog

The following must be specified:

NAME: A name or label identifying the host.

LOCATION: The host's location (selected from the popup list).

NUMBER OF LINKS: The number of physical access links connecting the host to the access switch.

LINK TYPE: The access link(s) type.

LINK SPEED: The access link(s) speed in bps (selected from the popup list).

PROTOCOL: The access link (s) protocol (selected from the popup list)

INTERFACE: Any host access equipment (eg PAD, FEP etc).

DEDICATED LINE(S): Select the radio button if the host has a leased access line.

Any host to host file transfers (if applicable) can be specified using the following fields:

PRIMARY DEST: The first destination host (selected from the popup list).

PRIMARY FLOW: The traffic flow between the host and its primary destination.

SECONDARY DEST: The second destination host (selected from the popup list).

SECONDARY FLOW: The traffic flow between the host and its secondary destination.

TRAFFIC UNIT: The traffic flow unit (selected from the popup list).

COMMENT: Any user comment pertaining to this host

- **DELETE host** — The user is prompted for a host name. That host is then deleted from the current working network scenario. If the subscriber database is re-saved to disk the host will be permanently deleted.

NOTE: The user will be forced to alter any subscribers that are currently homed to that host.

Geographic location

- **ADD Geographic location** — The user can add the details of a new geographic location using this command. Details are entered using the dialog box in figure 3.3.

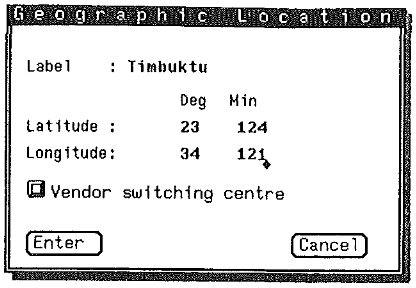

Figure 3.3 Geographic location entry dialog

The following must be specified:
LABEL: A name or label identifying the location.
LATITUDE DEG / MIN: The location's latitude.
LONGITUDE DEG / MIN: The location's longitude.
VENDOR SWITCHING CENTRE: Select the radio button if the location corresponds to a link vendor switching centre.
From now on, the new location will appear in all relevant popup lists.

- **DELETE Geographic location** — Opens a dialog box containing a popup list of all currently loaded geographic locations. The selected location is then deleted.
NOTE: The user is then forced to alter any network entity that was placed at that location.

Hardware profile

- **ADD Hardware profile** — The user is prompted for details on the new hardware component using the dialog box in fig 3.4.

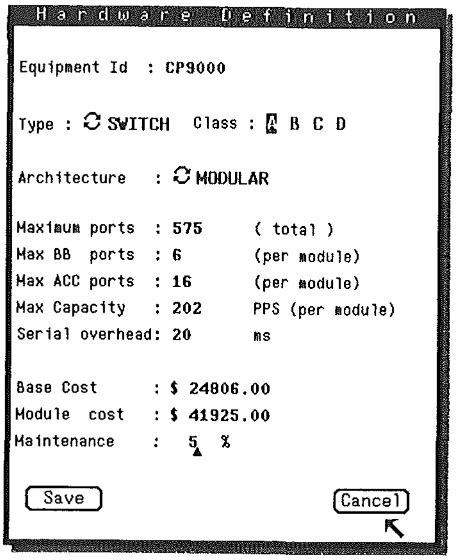

Figure 3.4 Hardware profile entry dialog

The following information is required:
EQUIPMENT ID: A unique label identifying the component. This label will appear in all hardware choice popup lists and menus.
TYPE: The equipment type: Switch, PAD, FEP etc (selected from the popup list).
CLASS: Assign an arbitrary class (A= high end component, D = entry level component).
ARCHITECTURE: The component's architecture (Modular or Fixed)
MAXIMUM PORTS: The maximum number of communication ports available.
MAX BB PORTS: The maximum number of backbone ports per module.

MAX ACC PORTS: The maximum number of access ports per module.

MAX CAPACITY: The maximum rated capacity (units dependant on equipment type).

SERIAL OVERHEAD: The load independent delay overhead (in ms).

BASE COST: The equipment purchase price (less the processing modules)

MODULE COST: The cost of each processing module.

MAINTENANCE COST: The cost associated with general maintenance. (expressed as a percentage of total value).

- **DELETE Hardware profile** — The hardware profile is selected from this hierarchical menu. The user will then be forced to modify any existing network component that relies on that hardware type.

Node

- **ADD Node** — New node details are entered using the dialog box in figure 3.5

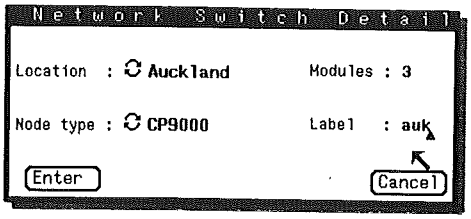

Figure 3.5 Node specification dialog

The following must be provided:

LOCATION: The node's geographic location (selected from the popup list).

MODULES: The number of processing modules (if the switch has a modular architecture).

NODE TYPE: The node equipment type (selected from the popup list).

LABEL: A 3-4 letter label that will identify this node within the topology window.

- **DELETE Node** — Opens a dialog box containing a popup list of all nodes in the current topology. Once one is chosen, it is permanently deleted from the current network.

NOTE: All backbone links terminating at that node are also deleted.

Backbone link

- **ADD Backbone link** — New backbone details are entered using the dialog box in figure 3.6

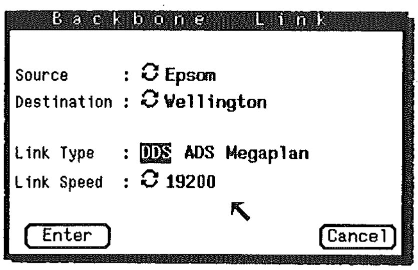

Figure 3.6 Backbone link specification dialog

The following must be provided:

SOURCE: The originating node (selected from the popup list).

DESTINATION: The destination node (selected from the popup list).

LINK TYPE: The backbone link type (selected from the popup list).

LINK SPEED: The link speed (selected from the popup list).

The new link is then added to the current topology.

- **DELETE Backbone link** Once the user has specified the target link using the resulting popup choice items, the link is permanently deleted from the current network topology.

Evaluation rule

Activates the rulebase editor (in figure 2.2). The user can add, delete or edit any rules from within this window.

Protocol definition

- **ADD Protocol definition** — A new protocol definition can be defined using this command. The details are entered by way of the dialog box in figure 3.7.

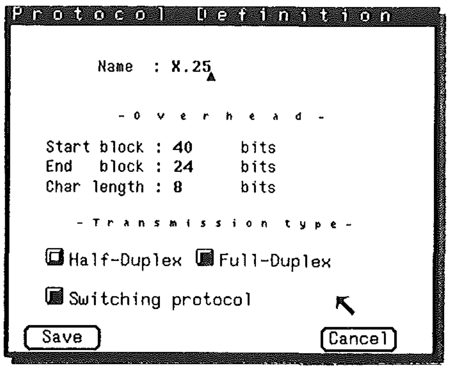

Figure 3.7 Protocol specification dialog

The following must be provided:
NAME: A name or label identifying the protocol.
START BLOCK: Start overhead in bits.
END BLOCK: End overhead in bits.
CHAR LENGTH: Character length in bits.
Once added the new protocol will appear in all relevant popup choice items.

- **DELETE Protocol definition** — The protocol definition is selected from the hierarchical menu. The user will then be forced to modify any existing network component that relies on that protocol definition.

MODIFY MENU

The modify menu contains commands for editing network entities.

Geographic location

Opens a dialog box containing a popup list of all currently loaded geographic locations. Once one is chosen its details can then be altered using the dialog box in figure 3.3.

Hardware profile

The user selects the hardware item from the hierarchical menu. The profile details can then be modified using the dialog box in figure 3.4

Protocol definition

The user selects the protocol from the hierarchical menu. The protocol definition can then be modified using the dialog box in figure 3.7

Terminal Centre

The user is prompted for the terminal centre's name. If it exists, the details can be modified using the dialog in figure 3.1

Host

The user is prompted for the host's name. If it exists, the details can be modified using the dialog in figure 3.2

Topology

Opens (if currently closed) the topology window and enables the user to modify various topological attributes using the menus detailed in the *topology window menus* section.

Evaluation rulebase

The user is prompted for the name of an evaluation rulebase. If it exists, it is loaded along with the editor (see figure 2.1). The user can then edit the rules from within this window.

REPORT MENU

The report menu contains the commands necessary for generating summary reports that detail the on-going design. In each case the user is prompted for a target file name. Once specified, the corresponding report is produced and written to that file.

Network profile report

Provides a detailed summary of the current network configuration. This report provides a summary of the nodes, backbone links and subscribers connected to the network.

Network subscriber report

Provides a detailed summary of all subscribers (i.e. terminal centres and hosts) that are connected to the network. The following details are provided for each subscriber: traffic submitted, location, response times, path length, host details, access costs.

Network backbone report

Provides a summary of the backbone links in the design. It provides the following information for each backbone link: link type, speed, delay, utilization and monthly cost analysis.

Network switch report

Generates a summary of nodes within the current design. The following information is provided on each backbone node: type, cost, location, delay, utilization, configuration (number of access/backbone ports, modules).

Network cost report

Generates a report detailing all network components and their respective cost breakdowns.

Network performance report

Provides a delay and utilization summary for all network components.

SNAPSHOT MENU

The commands in the *Snapshot* menu allow the user to save and restore any number of "network snapshots" during the design process. These snapshots allow the user to restore an earlier network configuration, if for instance a current path of exploration is becoming fruitless.

Take

The user will be prompted for a snapshot file name, defaulting to the last one used, and the current network configuration (e.g. topology, traffic, subscriber profiles etc.) saved. Any number of these "checkpoint" files can be made and later restored.

Restore

The user is prompted for a snapshot file name, again defaulting to the last one used. The exact network configuration at the time the snapshot was made is then restored.

NOTE: The current configuration, if not previously saved or detailed in another snapshot, will be permanently lost.

COST MENU

The *Cost* menu contains the commands for activating NDW's cost monitoring windows. All costs are subject to the parameters set using the *Costs* command in the *Parameters* menu.

Access link

Any subscriber's access link can be priced using this command. The user is prompted for the subscriber's name. If the subscriber exists, the access link cost components are then detailed in the window shown in figure 3.8

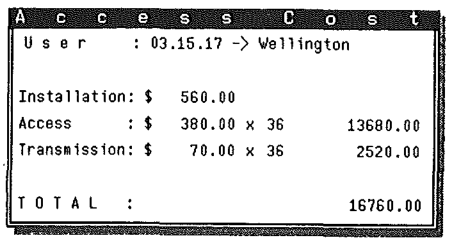

Figure 3.8 Access link cost window

Backbone link

Activates the backbone link cost window shown in figure 3.9.

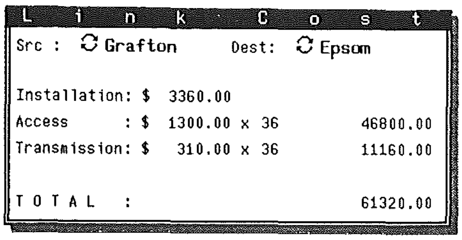

Figure 3.9 Backbone link cost window

The user can then select a backbone link using the popup choice items within the window. The link's cost components are then immediately displayed.

Node installation

Activates the node installation cost window shown in figure 3.10 .

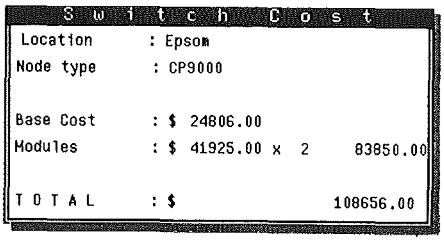

Figure 3.10 Node installation cost window

The user can then select a node using the popup choice items within the window. The node's cost components are then immediately displayed.

Network

The total cost of the current network configuration can be calculated and monitored using this command. Figure 3.11 details the network cost window. The total cost is broken down into its three components: the access network cost, the hardware cost, and the backbone network cost. The change in total cost since the last network alteration is also displayed under "delta cost".

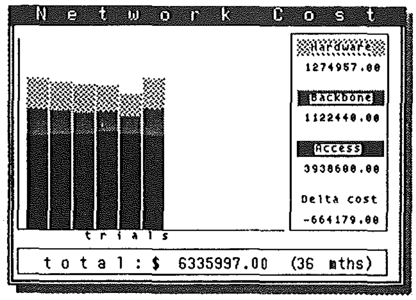

Figure 3.11 Network cost window

PERFORMANCE MENU

The *Performance* menu contains the commands for activating NDW's performance monitoring windows. Each plot or "trial" details the results or consequences of the user altering the network configuration .

Delay

- **Backbone link** — The user can graphically monitor the current delay on any backbone link using the window shown in figure 3.12

  The user can open any number of these windows. The selected links can also be dynamically changed using the src/destination popup choice items within each window.
- **Node** — Nodal delays can be monitored using the window in figure 3.13. Again, the user can open any number of these windows and monitor several nodes simultaneously.

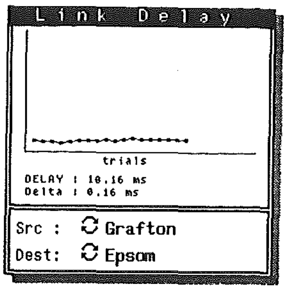

Figure 3.12

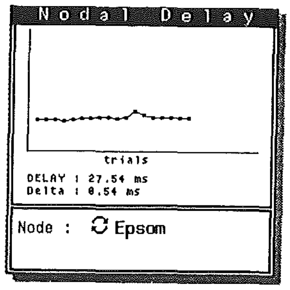

Figure 3.13

- **Network average** — The window in figure 3.14 is used to monitor the average network delay.

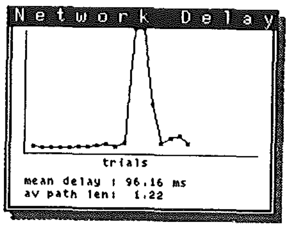

Figure 3.14 Average network delay window

The average virtual circuit path length is also displayed within this window.

- **User response time** — The user can monitor the user response time, less the host turn-round delay, for any network subscriber. The user is prompted for a subscriber's name; once found, the delay can be monitored using the window in Figure 3.15

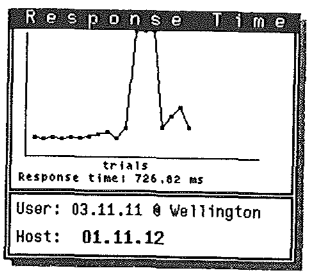

Figure 3.15 User response time window

The user can arbitrarily change the subscriber's destination host, and view the effect on response time by selecting a new host from the host popup choice within the window. NOTE: This only temporarily alters the destination . Any permanent changes must be made using the *Modify subscriber* command.

Utilization

- **Backbone link** The utilization of any number of backbone links can be monitored via the window in figure 3.16. The window displays the utilization history of the link as well as the actual flows (in PPS) over the link. The change in utilization between successive plots is also detailed within the window.

  The selected link can be changed using the src/dest popup choice items within each window.

- **Node** Individual node utilizations can be monitored using the window in figure 3.17. The selected nodes can be changed using the node popup choice item within each window.

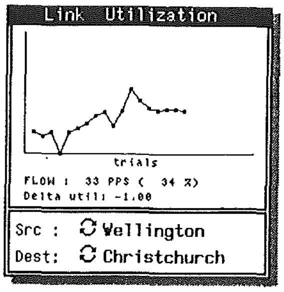

Figure 3.16

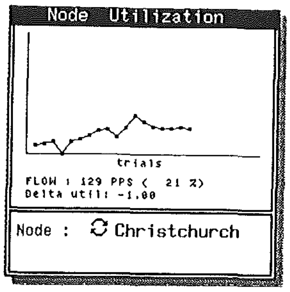

Figure 3.17

ATTAIN MENU

The user can use NDW to provide solutions to a range of problems. The user can ask NDW to "attain" these solutions using the commands in this menu. All solutions are displayed within the System messages window.

Node

- **Placement** — NDW will determine the most cost-effective location (given the current topology and subscriber base) to place a new node. All possibilities will be listed along with their corresponding costs.
- **Removal** — NDW will determine the best node to remove from the network. Again more than one possibility will be listed, so the final decision rests with the user.

Link

- **Placement** — NDW will suggest possible links to add given the current topology, traffic requirements, and design constraints. Each link is listed along with a preference factor: from 1 = highly recommended to 0.
- **Removal** — NDW will list several links that should be considered for removal.

Throughput

- **Increase** — NDW will determine the most cost-effective link and node to upgrade in order to increase the total network throughput.
- **Decrease** — NDW will determine the best link and node to downgrade in order to decrease the total network throughput and therefore decrease cost.

Delay decrease

NDW will determine the best component within the current network to upgrade, or possibly add, in order to decrease the current network's average delay.

Cost

- **Increase** — NDW will determine the most cost-effective component to upgrade or add to the current network.
- **Decrease** — NDW will determine the most costly and underutilized component within the current network configuration.

Connectivity

- **Increase** — NDW will determine the most suitable link to add to the network in order to increase its overall link connectivity.
- **Decrease** — NDW will find the best link to remove.

ENGINEER MENU

Re-evaluate network

Normally NDW will re-evaluate the network as the result of an alteration.This command, however, will force NDW to evaluate the current network using the currently loaded rulebase.

Tune backbone

The tune backbone command prompts the user for optimization criteria using the dialog box in figure 3.18

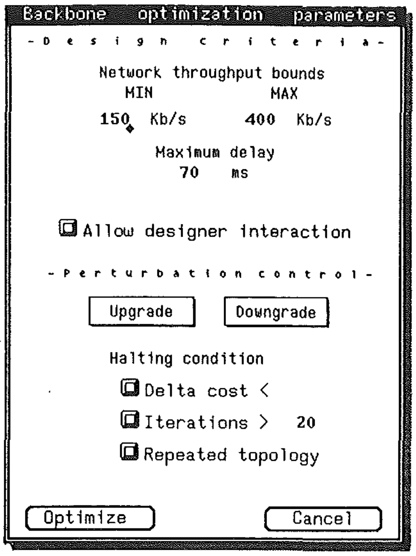

Figure 3.18 Optimization criteria dialog

The following must be specified:

THROUGHPUT MIN: The minimum network throughput (in Kbits/sec) at the delay specified in the maximum delay field below.

THROUGHPUT MAX: The maximum network throughput (in Kbits/sec) at the delay specified in the maximum delay field below.

MAXIMUM DELAY: The maximum average network delay.

Several other "perturbation control" parameters can also be specified using following buttons:

- **Upgrade** — The designer can alter the algorithm's link upgrade metrics via the *Upgrade* command. Alterations can be made using the window in figure 3.19

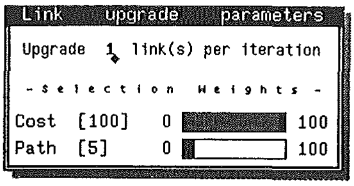

Figure 3.19 Upgrade settings window

SEL *x* LINKS: The number of links to upgrade during each iteration.

PATH: The weighting (i.e importance) given to a link's contribution to the average path length improvement

COST: The weighting given to a link's contribution to cost increase.

- **Downgrade** — The designer can alter the algorithm's link downgrade metrics via the *Downgrade* command. Alterations can be made using the window in Figure 3.20

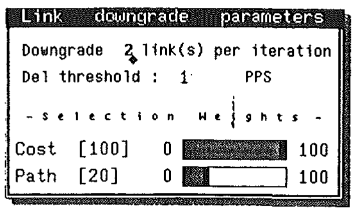

Figure 3.20 Downgrade settings window

SEL *x* LINKS: The number of links to downgrade during each iteration.

DEL THRESHOLD: Links with flows less than this threshold (in PPS) are given special consideration for deletion.

PATH: The weighting (i.e importance) given to a link's contribution to the average path length degradation

COST: The weighting given to a link's contribution to cost saving.

The user must also specify and enter one of the following halting conditions for the optimization process:

DELTA COST < x: Stop if the total cost saving is less than this figure.

ITERATIONS > x: Stop when the number of algorithm iterations surpasses this number.

REPEATED TOPOLOGY: Stop if the algorithm generates a previous topology.

HELP COMMAND

NDW provides the user with a comprehensive on-line help facility. The help command will activate the help window shown in figure 3.21.

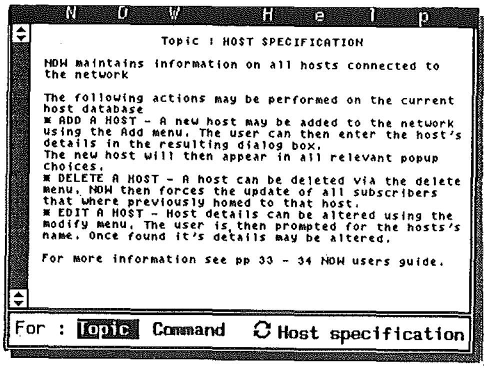

Figure 3.21 Help window

The user can obtain help for a particular command or topic by way of the *for* choice item. The corresponding topic or command can then be selected from the second popup list.

PARAMETERS MENU

The commands contained within NDW's *parameters* menu allow the user to set various operational settings and preferences.

NDW preferences

The following operational settings can be specified via the dialog in figure 3.22:

ENABLE AURAL FEEDBACK: Toggles system beeps during certain events (i.e. alerts etc)

ENABLE VISUAL FEEDBACK: Toggles the topology colour coding features.

ZOOM AREA: The topology zoom radius (in km) .

GRAPH DELTA: The monitor window graph x distance (in points) for successive plots.

AUTO LOAD: Checking and entering the names of these files will instruct NDW to automatically load them next time it is booted.

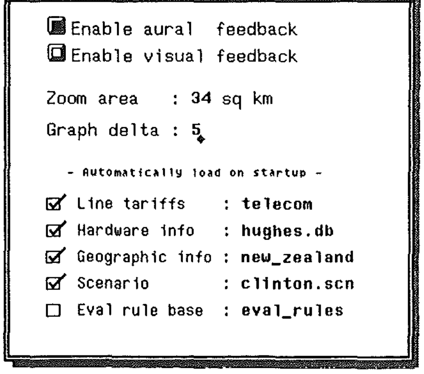

Figure 3.22 NDW preferences window

Costing

The following parameters can be specified using the dialog in figure 3.23:

RENTAL PERIOD: The rental period (in months) for any leased equipment. Any cost analysis involving leased equipment (i.e. communication links) will be calculated over this period of time.

USE NPV ADJUSTMENTS: NDW adjusts any economic analysis to Nett Present Value.

INTEREST RATE: The annual interest rate must be entered in this field if NPV adjustments are made.

ENABLE BACKGROUND CALCULATIONS: Toggles the constant recalculation and display of network costs after each user.

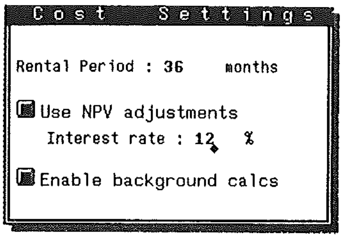

Figure 3.23 NDW cost settings window

Traffic profile

The current traffic profile can be altered using this command. The following attributes can be altered or specified using the window in figure 3.24.

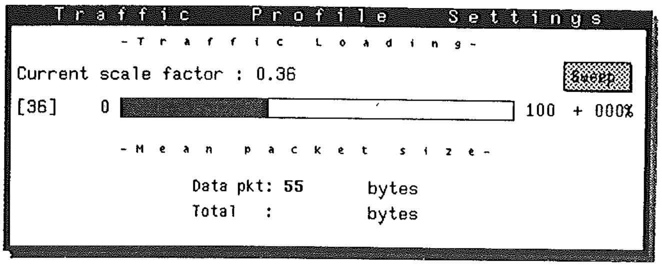

Figure 3.24 Traffic profile window

(1) The current traffic rate:

CURRENT SCALE FACTOR: Details the current global traffic scale factor, that is 2.34 = 2.34 times the traffic levels specified for each subscriber. The scale factor can be altered using the slider and popup choice item.

• **Sweep** — The sweep function displays the dialog box in figure 3.25. The user can specify a traffic scale range and halting condition. NDW will then automatically scale the traffic load over that range until one of the halting conditions is met.

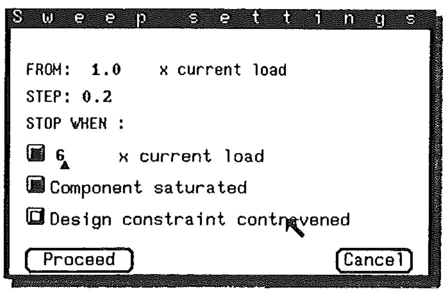

Figure 3.25 Traffic sweep window

The following must be specified:

START : The initial traffic scale factor

STEP : The scale step factor (i.e. 0.5 steps the scaling by 0.5 each time).

x X CURRENT LOAD : The halting scale factor value.

COMPONENT SATURATED : If checked, NDW will scale the traffic until a network component experiences infinite delay. The component(s) are then listed within the system messages window.

DESIGN RULE CONTRAVENED : If this is checked, NDW will scale the traffic until a design constraint is contravened.

(2) Packet size statistics:

DATA PKT: The mean size (in bytes) of a user data packet.

TOTAL: The overall mean packet size (in bytes) including network overhead.

Performance

The following performance settings can be specified via the window in figure 3.26

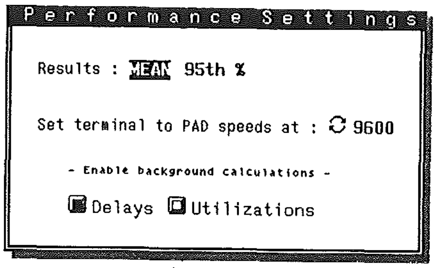

Figure 3.26 Performance settings window

RESULTS: Select the performance result type (mean or 95th percentile).

SET TERMINAL TO PAD SPEED: Sets the terminal to PAD speeds for all terminal centres.

ENABLE BACKGROUND CALCULATIONS: Toggles the constant recalculation and display of network delays and utilizations after each user action.

Design

The following design options and constraints can be set via the window in figure 3.27

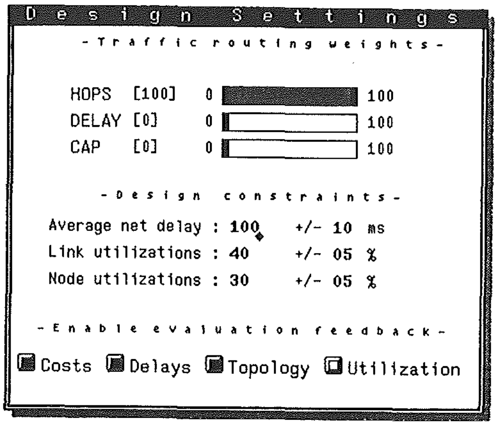

Figure 3.27 Design settings window

- NDW's traffic routing algorithm is based on calculating the shortest path using the weighted combination of:

  HOPS: The number of inter-nodal hops over a particular route.

  DELAY: The delay experienced over a route.

  CAP: The total link capacity over a route.

- Design constraints. NDW uses the following constraints for its link and node alteration suggestions:

  AV NET DELAY: The average network delay (in ms)

  LINK UTIL: The link utilization limit (% of capacity)

  NODE UTIL: The node utilization limit (% of capacity).

ENABLE BACKGROUND CALCULATIONS: Toggles the background re-evaluation of the network using a loaded rule-base.

Topology tuning

NDW's topology tuning system can be configured using this command. The user is presented with the same options discussed in the *Tune backbone* command description.

IMPORT MENU

External filters can be developed to extract traffic data from network management log files to either build or update subscriber profile files. Filter programs must reside in a directory called "NDW_filters" within the NDW directory. All filters in this directory will be listed in the *Import* menu and executed when they are selected from this menu.The user interacts with the filter programs by way of a popup window

All filter programs must conform to certain standards when writing to the subscriber profile file (see the document "Writing NDW filters" for more information).

HNS Filter

The HNS Filter allows the collation and importation of traffic data logged by the HUGHES Network System. See the document "HNS filter" for more details.

QUIT COMMAND

Selecting the *Quit* command on the main console will shut-down NDW. The user is asked to confirm that this is required and that all altered network data has been saved.

TOPOLOGY WINDOW MENUS

Add / Delete / Modify

- **Node** — The *Add, Delete* and *Modify* node commands within the topology window have the same functionality as their respective counterparts within the console window.
- **Backbone link** — Again the *Add, Delete* and *Modify* backbone commands within the topology window have the same functionality as their respective counterparts within the console window.

Zoom

The user is prompted for a node — once selected from the popup choice, NDW displays all network components, including the subscribers and their access network, that fall within an *x* km radius of that node. This radius can be set using the *zoom radius* field within the NDW preferences window.

Display

The user can set the following display options for the topology window using the window in figure 3.28

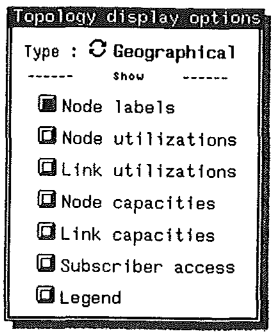

Figure 3.28 Topology display options window

TYPE: The topology display mode (Logical or Geographic).

NODE LABELS: Toggles the display of each node's label.

NODE UTILIZATION: NDW will colour code each node according to its current utilization.

LINK UTILIZATION: Backbone links are colour coded according to their current utilization.

NODAL CAPACITIES: Nodes are coded according to their capacity.

LINK CAPACITIES: Links are coded according to their capacity.

NODAL DELAYS : Nodes are colour coded according to their contribution to network delays.

LINK DELAYS : Links are colour coded according to their contribution to network delays.

SHOW LEGEND: NDW will display the colour coding legend.

LOGFILE ANALYSIS REPORTS

The logfile analysis programs discussed in Chapter eight produce the following reports on network activity.

MODULE TRAFFIC RATES

The traffic rates for every nodal port can be determined using the module utilization reports. The port protocol is also identified (X.25 or BackBone).

MODULE UTILIZATION REPORT
REPORT ON DATA PACKETS SENT/RECEIVED

Data from : 30/ 7/90
Node : HLZ (5)
Cluster : 1.1
Module : 1

HOURS	PORTS 1 X.25	2 X.25	3 X.25	4 X.25	5 BB	6 X.25	7 X.25	8 X.25
0 - 1	0	206	0	0	45	0	0	0
1 - 2	0	204	0	0	45	0	0	0
2 - 3	0	207	0	0	45	0	0	0
3 - 4	0	204	0	0	45	0	0	0
4 - 5	0	207	0	0	44	0	0	0
5 - 6	0	204	0	0	45	0	0	0
6 - 7	0	206	0	0	250	0	0	0
7 - 8	411	581	528	329	2633	188	0	0
8 - 9	1943	1491	0	6198	9408	12	1308	1124
9 - 10	1841	2863	3316	4160	7535	25	0	150
10 - 11	1071	2099	2049	9319	6498	2	566	43
11 - 12	465	4242	2201	33129	8628	878	0	180
12 - 13	1491	2847	989	32821	7763	2666	0	0
13 - 14	914	3558	2576	45088	13629	5420	0	0
14 - 15	796	4414	2258	43245	13591	3880	0	1604
15 - 16	776	2325	3331	34655	8956	1877	0	2367
16 - 17	1358	2282	1326	23548	7478	6058	0	153
17 - 18	329	3181	1366	697	1759	191	0	0
18 - 19	54	797	14	429	102	0	0	0
19 - 20	0	206	0	0	71	0	0	0
20 - 21	0	204	0	0	46	0	0	0
21 - 22	0	208	0	0	45	0	0	0
22 - 23	0	204	0	0	48	0	0	0
23 - 24	0	207	0	0	48	0	0	0

CLUSTER TRAFFIC RATES

The cluster utilization reports detail the nodal traffic levels for each PSC[1]. The utilization factor is based on the theoretical maximum processing rate published by Hughes [41].

CLUSTER UTILIZATION REPORT
REPORT ON DATA PACKETS SENT/RECEIVED

Data from: 30/ 7/90
Node : GFT (2)
Cluster : 1.1

	MODULES				
HOURS	LIM1	LIM2	LIM3	TOTAL	UTIL
0 - 1	828	0	750	1578	0.22 %
1 - 2	1179	0	707	1886	0.26 %
2 - 3	3830	0	736	4566	0.63 %
3 - 4	763	0	788	1551	0.21 %
4 - 5	782	0	800	1582	0.22 %
5 - 6	872	0	802	1674	0.23 %
6 - 7	1475	0	8445	9920	1.36 %
7 - 8	4866	7622	30795	43283	5.95 %
8 - 9	12311	29462	83846	125619	17.27 %
9 - 10	17217	36009	112175	165401	22.74 %
10 - 11	21083	35989	98910	155982	21.45 %
11 - 12	21726	38590	108957	169273	23.28 %
12 - 13	21220	23903	90266	135389	18.62 %
13 - 14	19057	23831	87535	130423	17.93 %
14 - 15	19730	49691	111655	181076	24.90 %
15 - 16	11281	34061	112408	157750	21.69 %
16 - 17	8471	16013	79244	103728	14.26 %
17 - 18	4322	8224	41963	54509	7.50 %
18 - 19	4130	1875	21825	27830	3.83 %
19 - 20	4018	2767	2519	9304	1.28 %
20 - 21	2579	518	2536	5633	0.77 %
21 - 22	2993	1376	4774	9143	1.26 %
22 - 23	6179	2025	5203	13407	1.84 %
23 - 24	6111	1211	2450	9772	1.34 %

[1] The Hughes architecture is detailed in figure 8.3 of Chapter eight.

PACKET SIZE SUMMARY

The hourly packet size distributions for any backbone port can be determined using the packet size report detailed below.

PACKET SIZE DISTRIBUTIONS (BBLINKS ONLY)

Data from : 30/ 7/90
Node : CHC (4)
Cluster : 1.5
Module : 1
Port : 1

	PACKET SIZE															
HOURS	1-16	17-32	33-48	49-64	65-80	81-96	97-112	113-128	129-144	145-160	161-176	177-192	193-208	209-224	225-240	241-256
0 - 1	39	216	91	36	50	0	0	0	0	0	22	2	160	14	0	0
1 - 2	37	234	85	36	50	0	0	0	0	0	4	2	158	14	0	0
2 - 3	39	180	84	54	50	0	0	0	0	0	2	2	158	14	0	0
3 - 4	38	192	103	36	50	0	0	0	0	0	66	2	158	14	0	0
4 - 5	36	184	65	66	50	0	0	0	0	0	24	2	158	14	0	0
5 - 6	35	180	111	36	50	0	0	4	0	0	4	2	160	14	0	0
6 - 7	398	234	84	36	50	0	0	0	0	0	2	2	160	14	0	0
7 - 8	1762	37	11	23	32	1	1	2	0	0	0	0	98	8	0	0
8 - 9	5523	115	44	14	11	14	8	165	0	1	21	0	0	0	3	235
9 - 10	11377	285	459	332	478	205	184	3532	0	0	0	0	0	0	0	0
10 - 11	12394	369	536	648	477	268	230	2919	0	0	8	0	0	0	0	56
11 - 12	13147	573	861	870	761	421	540	4064	13	1	1	5	7	1	7	222
12 - 13	8598	688	408	315	120	116	148	3378	0	0	0	0	0	0	0	0
13 - 14	11918	706	634	487	249	154	71	1526	19	3	5	21	6	10	3	225
14 - 15	18459	716	1487	1764	897	888	812	3999	0	1	12	2	0	1	0	117
15 - 16	15889	294	298	428	155	106	96	3146	0	0	4	0	0	0	0	40
16 - 17	10915	969	647	740	258	160	199	4727	4	15	8	8	4	5	1	197
17 - 18	3716	522	445	528	208	97	176	3549	0	1	15	0	1	3	2	154
18 - 19	1326	150	118	53	42	26	26	507	0	2	5	2	1	0	2	75
19 - 20	1136	349	344	181	52	41	74	992	0	0	0	3	0	0	0	0
20 - 21	427	15	24	16	4	6	12	495	0	0	0	3	0	0	0	0
21 - 22	1408	80	67	53	40	28	33	2636	0	0	0	3	0	0	0	0
22 - 23	163	45	62	25	15	7	18	211	0	0	0	3	0	0	0	0
23 - 24	39	3	0	0	0	0	0	0	0	0	0	3	0	0	0	0
TOTAL	118819	7336	7068	6777	4149	2538	2628	35852	36	24	203	67	1229	126	18	1321

DEVICE UTILIZATION REPORTS

The device utilization reports detail several operational statistics concerning the node's processor (PM) and line interface (LIM) modules namely:

- *CPU utilization* — Each module monitors its current CPU utilization and stores the average, high and low 'water marks' over the last hour.

- *Congestions status* — Modules periodically assess their congestion status; Nodes log the number of 100ms intervals the module was experiencing one of four states. (None = Uncongested, SL = Slight, Mod = Moderate, Crit = Critical.)

• *Data buffer queues* — Each module maintains a high and low water mark for their data buffer queues. These marks[2] reflect the buffer usage during the current statistics period.

MODULE UTILIZATION REPORT

Data from : 30/ 7/90
Node : EPS (1)
Cluster : 1.5
Module type : PSC Processing Module

HOUR	CPU UTIL(%)			CONGESTION(x100ms)				DATA Q UTIL(%)	
	Ave	High	Low	None	Sl	Mod	Crit	High	Low
0	20	34	0	35812	0	0	0	9.86	8.65
1	15	18	0	36189	0	0	0	10.31	8.65
2	8	17	0	35987	0	0	0	9.10	8.65
3	2	3	0	35986	0	0	0	8.65	8.65
4	2	3	0	36016	0	0	0	8.65	8.65
5	2	3	0	35996	0	0	0	8.65	8.65
6	14	33	0	35972	0	0	0	10.84	8.65
7	12	46	0	35796	0	0	0	11.89	8.65
8	34	60	0	35463	0	0	0	13.54	8.80
9	47	79	22	34869	0	0	0	14.30	9.10
10	42	66	0	35257	0	0	0	13.54	9.10
11	46	69	0	35026	0	0	0	12.49	9.10
12	41	70	0	35316	0	0	0	12.04	9.10
13	34	62	0	35534	0	0	0	12.04	9.10
14	38	73	0	35217	0	0	0	12.04	9.18
15	46	74	0	35075	0	0	0	13.09	9.33
16	56	76	29	34634	0	0	0	12.87	9.33
17	30	52	0	35701	0	0	0	11.36	8.88
18	13	31	0	36043	0	0	0	11.29	8.80
19	2	6	0	36067	0	0	0	9.03	8.88
20	6	45	0	36035	0	0	0	10.68	8.65
21	5	23	0	35935	0	0	0	9.56	8.58
22	13	28	0	35999	0	0	0	10.91	8.65
23	22	26	0	36050	0	0	0	9.63	8.65

BACKBONE TRAFFIC RATE

The backbone traffic report (overleaf) details the hourly traffic rates over all currently defined backbone links.

2 Percentage of total buffer space.

HNS INTEGRATED PACKET NETWORK

BACKBONE LINK UTILIZATION REPORT

REPORT ON DATA PACKETS SENT/RECEIVED

Data from : 30/ 7/90

Link											Hour													
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
EPS.11.16<->TKI.11.17	0	0	0	0	0	0	0	199	3324	1360	1488	1911	529	3825	2995	569	1907	2961	265	1	7	0	0	92
EPS.15.22<->GFT.11.32	137	122	131	156	163	168	4112	11649	29857	43786	33337	39561	32830	36182	38480	48284	35414	19912	11119	1047	817	2133	3855	827
EPS.15.21<->WLG.11.31	665	18291	8032	570	574	565	18300	16435	71186	118152	95710	111022	76864	101193	135601	120415	97851	51484	16532	6802	7332	10350	19292	32346
EPS.15.17<->NPL.11.27	44	43	43	44	44	44	42	720	2899	8528	5284	9310	9151	10753	10975	6552	3647	397	44	43	43	43	44	45
GFT.11.31<->AKL.11.31	612	586	605	632	637	634	4333	19111	53616	68564	65552	69338	57428	51368	72965	64304	44125	22026	10765	1476	1717	2615	1375	1626
WLG.11.32<->PMR.11.31	725	17859	8002	553	543	545	1380	10001	59285	54115	47232	58596	32725	47220	43575	43424	38400	34494	17235	8696	10150	12149	21172	32797
WLG.15.21<->CHC.15.11	892	1192	844	918	858	852	2023	5544	20376	52235	52271	58215	37540	47515	67704	57713	51679	24557	4796	6189	2243	7119	1074	376
WLG.15.22<->AKL.15.11	1508	1437	1418	1470	1437	1413	3341	14640	37974	40013	48836	55008	44370	45482	71550	62299	74911	40267	11522	849	929	838	830	939
WLG.19.22<->LHT.11.31	0	0	0	0	0	0	17768	9236	17633	21101	22851	31003	32826	29588	37735	28675	41361	24715	10529	51	110	385	51	52
WLG.21.11<->NAP.11.11	0	0	0	0	0	0	0	0	0	0	6501	17085	8871	10567	14733	20620	27241	10292	3110	533	260	46	49	49
CHC.11.16<->TIU.11.12	0	0	0	0	0	0	0	0	1921	4922	6175	4518	1652	2587	4500	2702	2368	3111	87	0	0	0	0	0
CHC.11.31<->DUD.11.31	205	199	1078	214	198	197	293	878	11857	34805	28775	41386	32678	30836	39390	34606	45346	10425	420	207	211	1896	206	4625
CHC.11.32<->INV.11.11	0	0	0	0	0	0	0	0	5462	10072	11789	21576	9939	11637	16570	17300	12528	2701	18	0	738	707	0	0
CHC.21.21<->AKL.21.11	1	0	0	0	0	1	60	5102	31570	26398	40438	41057	35875	24859	32654	35519	37157	8626	1144	710	672	620	672	684
HLZ.11.31<->AKL.11.32	1788	1752	1700	1801	1898	1842	6043	17613	47311	57353	58071	71564	68006	42002	57990	61777	47847	22123	12032	2333	1968	1926	1912	1846
HLZ.11.32<->ROT.11.11	135	130	129	132	135	130	259	4144	23180	18384	26087	41307	37536	20604	27416	29198	17486	3900	242	561	135	131	129	142
HLZ.11.15<->TRG.11.11	45	45	45	45	44	45	250	2633	9408	7535	6498	8628	7763	13629	13591	8956	7478	1759	102	71	46	45	48	48
HLZ.15.21<->PMR.15.11	727	716	712	730	719	721	721	1588	4412	2585	5035	12784	4949	5362	4103	4559	4371	1839	767	666	684	675	669	686
PMR.11.14<->LHT.11.14	55	52	49	53	55	53	49	66	102	68	54	61	72	76	55	57	86	59	85	0	0	0	0	2
NAP.11.15<->ROT.11.22	38	35	35	35	38	37	43	164	761	1858	2090	771	485	582	744	873	1232	551	45	39	38	36	35	38
TKI.11.21<->AKL.15.27	43	42	43	44	43	43	43	754	11250	17108	22388	18232	13031	18962	24588	18133	20450	2789	73	46	9670	8020	1823	46
DUD.11.15<->INV.11.21	41	40	41	45	41	41	40	78	75	47	41	48	44	46	44	53	52	44	41	45	44	41	40	47

appendix V

NET-X DESIGN TRANSCRIPTS

This appendix details the actions performed by the designer and the corresponding NDW responses that occured during the design of the networks detailed in Chapter ten.

Owing to NDW's highly interactive nature and to ensure brevity only the main events of the design process have been recorded.

The transcripts adopt the following notation:

- **NDW Menu Command (designer following a NDW suggestion).**
- **<u>NDW Menu Command (designer taking initiative).</u>**
- *NDW observation or suggestion made via the NDW system message window.*
- ***Feedback via the NDW monitor facility.***

ADS ACCESS DESIGN

The transcript below details the design using Analogue Data Service (ADS) access links and Digital Data Service (DDS) backbone links resulting in the cost dispersion graph in figure 10.2.

Designer Action	*NDW Feedback*	*Comment*
Load Scenario, **New Topology**		The current network is loaded and its topology cleared.
Attain node placement, **Add node** in Epsom (Eps)	*Best location to place a node is Epsom costing $5652789 (with 261 user ports)*	NDW lists all placement options. Placing a node at Epsom will result in the least cost single node network. This is mainly due to the number of hosts that would have zero access costs as a result of placing a node there. Add a CP 9000 Node
Attain node placement, **Add node** in Wellington (Wel)	*Best location to place a node is Wellington* *(111 user ports)*	Placing a node in Wellington will save $ 1224096 Add a CP 9000 Node
Attain link placement, **Add link:** Eps to Wel 128K	*Suggest adding:* *Eps to Wel 128K DDS*	What is the best link to add, given the design constraints in Chapter 11.
Attain node placement, **Add node** in Christchurch (Chc)	*Best location to place a node is Christchurch saving $572439 (51 user ports)*	Add a CP 9000
Attain link placement, **Edit link :** Eps to Well 48K **Add link :** Eps to Chc 48K Wel to Chc 9K6	*Suggest adding/changing the following links:* *Eps to Well 48K DDS link* *Eps to Chc 48K DDS link* *Wel to Chc 4K8 DDS link*	

Attain node placement, **Add node** in Hamilton (Ham)	*Best location to place a node is Hamilton saving $335534 (58 user ports)*	Add a CP 9000
Attain link placement, **Add link :** Eps to Ham 48K Well to Ham 9K6	*Suggest adding the following links:* *Eps to Ham 48K DDS link* *Well to Ham 9K6 DDS link*	
Attain node placement, **Add node** in Dunedin (Dud)	*Best location to place node is Dunedin saving $232543 (18 user ports)*	Add a CP 9000
Attain link placement, **Add link :** Chc to Dud 48K	*Suggest adding the following link:* *Chc to Dud 48K DDS link*	
Attain node placement, **Add node** in Palmerston North (Pmr)	*Best location to place a node is Palmerston North saving $165455 (21 user ports)*	Add a CP 9000
Attain link placement, **Add link :** Eps to Pmr 19K2 Wel to Pmr 19K2	*Suggest adding the following links:* *Wel to Pmr 19K2 DDS link* *Eps to Pmr 19K2 DDS link*	
Attain node placement, **Add node** in Rotorua (Rot)	*Best location to place a node is Rotorua saving $182734 (21 user ports)*	Add a CP 9000
Attain link placement, **Add link :** Eps to Rot 19K2 **<u>Add link</u> :** Rot to Pmr 9K6 **<u>Edit link</u> :** Alter Pmr - Rot link to Wel - Rot (still 9K6) **Edit link :** Eps to Rot 48K **Link utilization:** Wel to Rot	*Suggest adding the following link:* *Eps to Rot 19K2 DDS* *Failure of the Eps to Rot link will detach components* *Pmr to Rot Link utilization now 0% suggest deletion of link* *Eps to Rot Link utilization now 41% upgrade:* *Eps to Rot link to 48K DDS* ***Wel to Rot link utilization now 13%***	Add Rot to Pmr to satisfy connectivity constraint. Addition of the links resulted in redistribution of traffic flows resulting in: • an overutilized link • an underutilized link NDW detected and suggested solutions to these problems
Attain node placement, **Add node** in Napier (Nap)	*Best location to place node is Napier saving $75851 (6 user ports)*	Add a CP 9724 (this node is only serving 6 subscribers)
Attain link placement, **Add link :** Eps to Nap 9K6 **<u>Add link</u> :** Rot to Nap 9K6 **Edit link :** Rot to Nap 19K2	*Suggest adding the following link:* *Eps to Nap 9K6 DDS link* *Failure of the Eps to Nap link will detach components* *Rot to Nap Link utilization now 38% upgrade:* *Rot to Nap link to 19K2 DDS*	Add Rot to Nap to satisfy connectivity constraint.
Attain node placement, **Add node** in Tauranga (Tng)	*Best location to place a node is Tauranga saving $36971 (6 user ports)*	Add a CP 9724 (this node is only serving 6 subscribers)
Attain link placement, **<u>Add link</u>:** Trg to Ham 9K6 **<u>Cost link</u>:** Trg to Ham **<u>Edit link</u>:** Alter Trg - Ham link to Rot - Trg (still 9K6)	*Suggest adding the following links:* *Eps to Tng 4K8 DDS link* *Ham to Tng 4K8 DDS link* ***Link cost $ 19640*** ***Link cost $ 16760***	Designer finding best link to add Not much difference in link costs, however Tng to Ham link results in a more reliable topology
Attain node placement, **Add node** in New Plymouth	*Best location to place node is New Plymouth saving $24682 (7 user ports)*	Add a CP 9724 (this node is only serving 7 subscribers)

Attain link placement, **Add link :** Ham to Npl 9K6	*Suggest adding the following links:* *Ham to Npl 4K8 DDS link* *Eps to Npl 4K8 DDS link*	NDW suggest s 2 possible links, however it also indicates that the Ham to Npl is twice as preferable to the Eps to Npl Link.
Attain node placement, **Add node** in Lower Hutt (Lht)	*Best location to place node is Lower Hutt saving $24448 (10 user ports)*	Add a CP 9724 (this node is only serving 10 users)
Attain link placement, **Add link :** Wel to Lht 9K6	*Suggest adding the following links:* *Eps to Lht 9K6 DDS link* *Wel to Lht 4K8 DDS link*	
Attain node placement, **Add node** in Grafton (Gft)	*Best location to place node is Grafton saving $3746 (55 user ports)*	Add a CP 9000 (this node will become one of the three nodes in this area)
Attain link placement, **Edit link:** Alter Wel - Eps link to Wel - Gft (still 48K) **Add link :** Eps to Gft 19K2 **Add link :** Rot to Gft 19K2 **Edit link:** Eps - Gft 64K **<u>Network Delay</u>** **<u>Edit link</u>:** Alter Rot - Gft link to Rot - Eps (still 19K2)	*Suggest adding / changing the following links:* *Wel to Gft 19K2 DDS link* *Eps to Gft 19K2 DDS link* *Ham to Gft 19K2 DDS link* *Eps to Gft Link utilization now 94% upgrade:* *Eps to Gft link to 64K DDS* ***Average path length 1.03*** ***Average path length 0.97***	 Designer asking the question "would it be better if the Rotorua node was connected to the Epson node" ...
Attain node placement, **<u>Add node</u>** in Auckland (Akl)	*Placing a node in Auckland will service 55 subscribers*	Placement of nodes no longer outweighs the saving in access costs. This location was chosen because: • There are 55 user ports including 16 hosts connected to the Auckland node • It would relieve some of the load from the Grafton node. • A third node in the Auckland region was needed for reliability.

Attain link placement, **Add link :** Eps to Akl 9K6 Gft to Akl 9K6 **Edit link:** Eps - Akl 48K Akl - Gft 48K Wel - Chc 19K2 <u>**Network Delay**</u> <u>**Edit link**</u>: Alter Wel - Akl link to Wel - Gft (still 19K2) <u>**Edit link**</u>: Alter Wel - Gft link to Wel - Akl (still 19K2)	*Suggest adding/ changing the following links:* *Gft to Akl 9K6 DDS link* *Eps to Akl 9K6 DDS link* *Wel to Akl 9K6 DDS link* *Eps to Akl Link utilization now 82% upgrade:* *Eps to Akl link to 48K DDS* *Gft to Akl Link utilization now 122% upgrade:* *Gft to Akl link to 48K DDS* *Wel to Chc Link utilization now 57% upgrade:* *Wel to Chc link to 19K2 DDS* ***Average network delay 56ms*** ***Average path length 1.12*** ***Average network delay 121ms*** ***Average path length 1.09***	 Designer asking the question "would it be better if the Wellington node was connected to the Grafton node instead of the Auckland Node" ...
Attain node placement, <u>**Add node**</u> in Invercargill	Placing a node in Invercargill will service 5 subscribers	Add a CP 9724 (this node is only serving 5 subscribers)
Attain link placement, **Add link :** Inv to Chc 19K2 **Edit link :** Wel to Chc 48K	*Suggest adding the following links:* *Inv to Chc 19K2 DDS link* *Network throughput has dropped to 168Kb/s upgrade:* *Wel to Chc link to 48K DDS*	
Attain node placement, <u>**Add node**</u> in Timaru	Placing a node in Timaru will service 4 subscribers	Add a CP 9724 (this node is only serving 4 subscribers)
Attain link placement, **Add link :** Tim to Chc 9K6	*Suggest adding the following links:* *Tim to Chc 9K6 DDS link*	
Attain node placement, <u>**Add node**</u> in Tamaki	Placing a node in Tamaki will service 6 subscribers	Add a CP 9724 (this node is only serving 6 subscribers)
Attain link placement, **Add link :** Akl to Tki 9K6	*Suggest adding the following links:* *Akl to Tki 9K6 DDS link*	

DDS ACCESS DESIGN

The transcript below details the design using Digital Data Service (DDS) access links and DDS backbone links resulting in the cost dispersion graph in figure 10.1.

Designer Action	*NDW Feedback*	*Comment*
Load Scenario, **New Topology**		The current network is loaded and its topology cleared.

Attain node placement, **Add node** in Epsom	*Best location to place a node is Epsom*	NDW lists all placement options. Placing a node at Epsom will result in the least cost network. This is mainly due to the number of hosts that would have zero access costs as a result of placing a node there.
Attain node placement, **Add node** in Wellington	*Best location to place a node is Nelson*	Placing a node in Nelson will save $961670, however Wellington is a better choice (main centre) saving $940790
Attain link placement, **Add link:** Eps to Wel 128K	*Suggest adding:* *Eps to Well 128K DDS*	What is the best link to add, given the design constraints (see Chapter 11)
Attain node placement, **Add node** in Christchurch	*Best location to place a node is Timaru*	Placing a node in Timaru will save $299435, however Christchurch is a better choice (main centre) saving $297635
Attain link placement, **Edit link :** Eps to Wel 48K **Add link :** Eps to Chc 48K Well to Chc 9K6	*Suggest adding/changing the following links:* *Eps to Well 48K DDS link* *Eps to Chc 48K DDS link* *Well to Chc 4K8 DDS link*	
Attain node placement, **Add node** in Dunedin	*Best location to place a node is Dunedin*	
Attain link placement, **Add link :** Chc to Dud 19K2	*Suggest adding:* *Chc to Dud 19K2 DDS*	
Attain node placement, **Add node** in Hamilton	*Best location to place a node is Hamilton*	
Attain link placement, **Add link :** Eps to Ham 48K Wel to Ham 9K6	*Suggest adding/changing the following links:* *Eps to Ham 48K DDS* *Wel to Ham 9K6 DDS*	
Attain node placement, **Add node** in Grafton	*Best location to place a node is Grafton*	Grafton suggested because of the multiple hosts at that site (their access costs becomes zero if a node is placed here)
Attain link placement, **Add link :** Gft to Ham 19K2 Eps to Gft 19K2 **Edit link :** Eps to Gft 48K	*Suggest adding/changing the following links:* *Gft to Ham 19K2 DDS* *Eps to Gft 19K2 DDS* *network average delay now 99.5 ms upgrade:* *Eps to Gft link to 48K DDS*	Addition of the links resulted in redistribution of traffic flows. As a consequence the initial Eps to Gft link size suggested by NDW became saturated. Using the rule base NDW detected this and suggested the link should be upgraded to accommodate the new flows.
Attain node placement, **Add node** in Palmerston North	*Best location to place a node is Palmerston North*	
Attain link placement, **Add link :** Eps to Pmr 9K6 Wel to Pmr 19K2 **Edit link :** Wel to Ham 48K	*Suggest adding/changing the following links:* *Eps to Pmr 9K6 DDS* *Wel to Pmr 19K2 DDS* *Wel to Ham link delay now 66 ms upgrade:* *Wel to Ham link to 48K DDS* *Node utilization now 35% upgrade Epsom node*	Addition of the links resulted in redistribution of traffic flows which saturated an existing link NDW detected this problem and suggested upgrade options NDW also drew attention to the excessive load on the Epsom Node (35%). This was ignored at the present time.
Attain node placement, **Add node** in Rotorua (Rot)	*Best location to place a node is Rotorua*	

Attain link placement, **Add link :** Gft to Rot 19K2 **Add link :** Rot to Pmr 9K6 **Edit link :** Eps to Gft 64K	*Suggest adding the following link: Gft to Rot 19K2 DDS* *Failure of the Gft to Rot link will detach components* *Eps to Gft link utilization now 44% upgrade: Eps to Gft link to 64K DDS*	Add Rot to Pmr to satisfy connectivity constraint. Addition of the links resulted in redistribution of traffic flows resulting in: • an overutilized link NDW detected and suggested solutions to these problems
Attain node placement, **Add node** in Napier (Nap)	*Best location to place a node is Napier*	NDW indicated that placing a node in either Napier or Gisbourne would save $ 18995.00 Napier was chosen because it corresponded to node location in the current NET-X topology.
Attain link placement, **Add link :** Rot to Nap 9K6	*Suggest adding the following link: Eps to Nap 9K6 DDS*	The link suggestion is based on the traffic requirement between the source and destination.
Attain node placement, **Add node** in Auckland (Akl)		Placement of nodes no longer outweighs the saving in access costs. This location was chosen because: • There are 16 hosts connected to the Auckland node • It would relieve some of the load from the Grafton node. • A third node in the Auckland region was needed for reliability.
Attain link placement, **Add link :** Gft to Akl 9K6 Eps to Akl 9K6 **Edit link :** Alter Wel - Eps link to Wel - Akl (still 48K) **Edit link :** Gft to Akl 48K **Edit link :** Wel to Chc 48K **Network delay** **Edit link :** Alter Gft - Rot link to Akl - Rot (still 19K2)	*Suggest adding/changing the following links:* *Gft to Akl 9K6 DDS* *Eps to Akl 9K6 DDS* *Wel to Akl 9K6 DDS* *Link overutilized upgrade: Gft to Ak link to 48K DDS* *Wel to Chch link is overutilized at 115% suggest upgrading to 48K* ***Av path length : 0.97*** ***Av path length : 0.92***	Addition of the links resulted in redistribution of traffic flows which resulted in the following problems: • over utilized links • underutilized links Chch - Akl traffic routed through Wellington now. Designer wondering if it would be better to connect the Rot node to the Akl node?
Attain node placement, **Add node** in Invercargill	Placing a node in Invercargill will service 5 subscribers	Add a CP 9724 (this node is only serving 5 subscribers)
Attain link placement, **Add link :** Chc to Inv 19K2	*Suggest adding the following links:* *Inv to Chc 19K2 DDS link*	
Attain node placement, **Add node** in New Plymouth	Placing a node in New Plymouth will service 7 subscribers	Add a CP 9724 (this node is only serving 7 subscribers)
Attain link placement, **Add link :** Ham to Npl 9K6	*Suggest adding the following links:* *Ham to Npl 4K8 DDS link*	
Attain node placement, **Add node** in Lower Hutt	Placing a node in Lower Hutt will service 10 subscribers	Add a CP 9724 (this node is only serving 10 subscribers)
Attain link placement, **Add link :** Wel to Lht 9K6	*Suggest adding the following links:* *Wel to Lht 4K8 DDS link*	
Attain node placement, **Add node** in Timaru	Placing a node in Timaru will service 4 subscribers	Add a CP 9724 (this node is only serving 4 subscribers)

Attain link placement, **Add link :** Chc to Tim 9K6	*Suggest adding the following links:* *Chc to Tim 4K8 DDS link*	
Attain node placement, **Add node** in Tauranga	Placing a node in Tauranga will service 6 subscribers	Add a CP 9724 (this node is only serving 6 subscribers)
Attain link placement, **Add link :** Ham to Tng 9K6	*Suggest adding the following links:* *Ham to Tng 4K8 DDS link*	
Attain node placement, **Add node** in Tamaki	Placing a node in Tamaki will service 6 subscribers	
Attain link placement, **Add link :** Akl to Tki 9K6	*Suggest adding the following links:* *Akl to Tki 9K6 DDS link*	

appendix VI

SUPERNET DESIGN

In addition to the design of the NET-X network detailed in sections IV and V of this thesis, NDW has been employed for the redesign of a New Zealand government X.25 packet switching network.

Because of its confidential nature, the network will be referred to in this appendix under the assumed name "SUPERNET".

SUPERNET's current topology consists of a highly meshed structure[1] comprised of 48K backbone links. Surprisingly, this topology was the result of a three year old design and it has remained unchanged since then. Since the reorganisation of government departments, the network operators have now become accountable for the running costs of the network, thus spurring the need for the networks re-evaluation.

The parameters for the design were:

- Maximum link utilization over the peak 5 minutes is 35%;
- Each node is two-connected;
- Maximum average network delay of 80ms;
- Average packet size 55 bytes;
- Maximum hop count resulting from a single link failure is four;
- Nodal throughput 1500 PPS.

SUPERNET was equipped with network monitoring facilities which enabled the collection of traffic data.

NDW was able to highlight excessive link costs and suggest a more cost effective configuration that met the stated design requirements.

The design results are detailed in Table VI.a.

---

[1] SUPERNET's existing structure is detailed in figures 6.6 and 6.7 of Chapter six.

Attribute	Initial design	NDW design
Topology		
Number of switches	16	16
Number of links	28	22
Performance		
Average network delay (ms)	59.84	54.60
Average path length (hops)	1.32	1.07
Throughput (Kb/s @ 80ms)	203.00	240.00
Annual Cost		
Hardware	$ 50433.00	$ 42466.00
Backbone network	$ 466560.00	$ 356400.00
Access network	$ 0.00	$ 0.00
TOTAL ANNUAL COST	$ 516993.00	$ 398866.00

Table VI.a SUPERNET design comparison

The main design savings resulted from the elimination of six redundant 48K backbone links and the reconfiguration of several existing 48K links.

The new structure represents a yearly cost saving of 23% or $118,127.00 over the existing configuration and exhibits an 18% increase in traffic handling capacity.

BIBLIOGRAPHY

[1] C. ANTUNES, J. CLIMACO, ET AL. "An Interactive Computer Aided Planning Package for Rural Telephone Networks", *Proceedings of the 12th International Teletraffic Congress*, Torino, June 1988, pp. 3.2B.6.1-3.2B.6.7.

[2] J. AYOUB. "Reduction of Redundant Links in Communication Networks", *Proceedings of the Eight International Conference on Computer Communications*, September 1986, pp.759-763.

[3] D. BACHMANN, M. SEGAL, ET AL. "NetMod: A Design Tool for Large-Scale Heterogeneous Campus Networks", *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 1, January 1991, pp. 15-24.

[4] S.H. BAKRY, B. EL-REDAISY, ET AL. "Computer simulation of a packet switching computer network", *Computer Communications*, vol. 11, no. 3, June 1988, pp. 124-129.

[5] P.C. BELL AND R.M. O'KEEFE. "Visual Interactive Simulation - History, recent developments, and major issues", *Simulation* , vol. 37, no. 3, 1987, pp. 109-116.

[6] BGS SYSTEMS, "BEST/1", *product description*, 128 Technology Centre, Waltham, Massachusetts, USA, 1988

[7] R. BOORSTYN AND H. FRANK. "Large-Scale Network Topological Optimization", *IEEE Transactions on Communications*, vol. COM-25, no. 1, January 1977, pp. 29-47.

[8] A. BURATTIN, U. MAZZEI, ET AL. "Computer-Aided Planning of Public Packet-Switched Networks", *Computer Networks and ISDN Systems*, vol. 10, no. 3, 1985, pp. 175-183.

[9] J. BUZEN. "Design of Modeling Tools for Computer Capacity Planning" in *Modelling Techniques and Tools for Performance Analysis '85*, Ed. N. Abu El Ata, Elsevier Science Publishers, North-Holland, 1986, pp. 7-11.

[10] R. CAMERON. "The Planning and Managing of Packet-Switching Networks", *Data Communications*, vol. 15, no. 2, March 1986, pp. 227-232.

[11] CCITT. "Blue book, recommendation Q.544", Fascicle VI.5, 1988.

[12] CCITT. "Telephone operation: Quality of Service and Tariffs." in *Orange Book*. Vol. II.2. International Telecommunications Union, Geneva, 1979.

[13] W. CHOU. "ACK/TOPS - An Integrated Network Design Tool", *IEEE International Conference on Communications*, 1981, pp. 4.1.1-4.1.7.

[14] W. CHOU. *Computer Communications Volume I: Principles*, Prentice Hall, 1985.

[15] W. CHOU, "Optimization of Data/Computer Networks" in *Computer Communications Volume II: Systems and Applications*, Prentice Hall, 1985.

[16] W. CHOU AND H. FRANK. "Routing Strategies for Computer Network Design", *Proceedings of the Symposium on Computer-Communications Networks and Teletraffic*, Brooklyn, April 1972, pp. 301-309.

[17] W. CHOU AND D.L. SAPIR. "A Generalized Cut-Saturation Algorithm for Distributed Computer Communications Network Optimization", *Proceedings of the 1982 ICC*, June 1982, pp. 4C.2.1-4C.2.6.

[18] G.J. COVIELLO AND R.D. ROSNER. "Cost Considerations for a Large Data Network", *Proceedings of the 2nd International Conference on Computer Communications*, Stockholm Sweden, August 1974, pp. 289-294.

[19] D. DAVIES. "The Control of Congestion in Packet-Switching Networks", *IEEE Transactions on Communications*, vol. COM-22, no. 11, June 1972, pp. 546-550.

[20] P. DENNING AND J. BUZEN. "The Operational Analysis of Queueing Network Models", *ACM Computing Surveys*, vol. 10, no. 3, September 1978, pp. 225-261.

[21] H. DYSART. "NEWCLUST: An Algorithm for the Topological Design of Two-Level, Multidrop Teleprocessing Networks", *IEEE Transactions on Communications*, vol. COM-26, no. 1, January 1978, pp. 55-62.

[22] L. ESAU AND K. WILLIAMS. "A Method for Approximating the Optimal Network", *IBM System Journal*, vol. 5, no. 3, 1966, pp. 142-147.

[23] E. FELDMAN AND F. LEHNER. "Warehouse Location User Continuous Economics of Scale", *Management Science*, vol. 12, May 1966, pp. 670-684.

[24] J. FERNANDEZ AND D. LIDDY. "Don't just guess: How to figure delays in private packet networks", *Data Communications*, vol. 17, no. 3, March 1988, pp. 197-210.

[25] R.W. FLOYD. "Algorithm 97: Shortest Path", *Communications of the ACM*, vol. 5, 1962, p 345.

[26] H. FRANK AND W. CHOU. "Topological Optimization of Computer Networks", *Proceedings of the IEEE*, vol. 60, no. 11, November 1972, pp. 1385-1397.

[27] H. FRANK AND I. FRISCH. "Network Analysis", *Scientific American*, vol. 223, July 1970, pp. 94-103.

[28] H. FRANK, I. FRISCH, ET AL. "Topological Considerations in the Design of the ARPA Computer Network", *Proceedings of the Spring Joint Computer Conference*, Washington DC, 1970, pp.581-587.

[29] M. GERLA, H. FRANK, ET AL. "A cut-saturation algorithm for topological design of packet-switched communications networks", *Proceedings of the National Telecommunications Conference*, San Diego CA, December 1974, pp. 1704-1085.

[30] M. GERLA AND L. KLEINROCK. "On the Topological Design of Distributed Computer Networks", *IEEE Transactions on Communications*, vol. COM-25, no. 1, January 1977, pp. 48-60.

[31] V.M. GROUT. "*Optimisation Techniques for Telecommunication Networks*", Ph.D Thesis, Department of Communication Engineering, Faculty of Technology, Plymouth Polytechnic, UK, 1988.

[32] J. GUERINEAU, M. DAO, ET AL. "OTARIE: An Interactive Tool for Performance Evaluation Studies in Telephone Networks", *Proceedings of the 12th International Teletraffic Congress*, Torino, June 1988, pp. 3.2B.5.1-3.2B.5.7.

[33] R. GUNNARSSON AND K. NIVERT. "Optimum Grade of Service in Telecommunication Networks", *Telecommunication Journal*, vol. VII, no. 51, 1984, pp. 323-326.

[34] E. HANSLER, G. MCAULIFFE, ET AL. "Optimizing the Reliability in Centralized Computer Networks", *IEEE Transactions on Communications*, vol. COM-20, no. 3, June 1972, pp. 640-644.

[35] R. HARRIS. "MINDER: An Interactive Planning Tool for Network Planners", *Proceedings of the 12th International Teletraffic Congress*, Torino, June 1988, pp. 3.2B.4.1-3.2B.4.7.

[36] G. HELD. "Why taxes matter in selecting equipment" in *Data Network Design Strategies*, Ed. R. Sarch, McGraw Hill, New York, 1983, pp. 264-268.

[37] P. HICKS. "Throughput and Delay Performance Measurement of Packet Switching Data Networks", *IEEE Global Telecommunications Conference & Exhibition*, Hollywood Florida, November 1988, pp. 0527-0534.

[38] HTL TELEMANAGEMENT LTD. "Network Tools For Design", *product description*, 3901 National Drive, Burtonsville, Maryland USA, 1989.

[39] HUGHES NETWORK SYSTEMS INC. "Hughes Automated Network Design System (HANDS)", *product description*, 11717 Exploration Lane, Germantown, Maryland, USA, 1989.

[40] HUGHES NETWORK SYSTEMS INC. "Hughes IPN Tape Format Reference Manual", *product manual,* 1990.

[41] HUGHES NETWORK SYSTEMS INC. "Integrated Packet Network System Architecture Manual", *product manual,* 1987.

[42] HUGHES NETWORK SYSTEMS INC. "IPN Network Control Processor (NCP) Operator's Reference Manual", *product manual,* 1989.

[43] HUGHES NETWORK SYSTEMS INC. "IPN Product Description (Revision 3.2)", *product manual,* 1989.

[44] HUGHES NETWORK SYSTEMS INC. "NMR Facility Operator's Procedures Manual", *product manual,* 1989

[45] R. HUNT. "Performance analysis of a large wide area communications network", *Presented at Computer Communications '85 ASIA,* Singapore, December 1985, pp. 79-95.

[46] H. INOSE AND T. SAITO. "Theoretical Aspects in the Analysis and Synthesis of Packet Communication Networks", *Proceedings of the IEEE,* vol. 66, no. 11, November 1978, pp. 1409-1422.

[47] P. JACKSON AND C. STUBBS. "A Study of Multiaccess Computer Communications", *AFIPS Conference Proceedings,* 1969, pp.491-504.

[48] S.P. JAIN AND K. GOPAL. "On Network Augmentation", *IEEE Transactions on Reliability,* vol. R-35, no. 5, December 1986, pp. 541-543.

[49] R. JAN AND F. HWANG. "Topological Optimization Problem of Communication Networks Subject to a Reliability Constraint", *INFOCOM '90,* 1990, pp. 487-494.

[50] H. JUCHTER AND R. LEHNERT. "NETCON- An Integrated Tool for Planning and Configuration of Packet Switching Networks", *Proceedings of the 8th International Conference on Computer Communications,* September 1986, pp. 741-746.

[51] B.W. KERNIGHAN AND D.M. RITCHIE. *The C Programming Language,* Prentice-Hall Inc., Englewood Cliffs New Jersey, 1978.

[52] H. KITAJIMA AND K. OHMACHI. "Integrated Tool for Computer Network System Configuration Planning." in *Teletraffic Issues in an Advanced Information Society,* Ed. M. Akiyama, Elsevier Science, North-Holland, 1985, pp. 972-977.

[53] L. KLEINROCK. "Analytic and Simulation Methods in Computer Network Design", *Spring Joint Computer Conference,* 1970, pp. 569-579.

[54] L. KLEINROCK. *Queueing Systems : Computer Applications,* Vol. 2., Wiley, New York, 1976.

[55] H. KOBAYASHI. *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology,* Addison-Wesley, 1978.

[56] B.R. KONSYNSKI AND W.E. BRACKER. "Defining requirements for a computer-aided net-design package" in *Data Network Design Strategies*, Ed. R. Sarch , McGraw Hill, New York, 1983, pp. 106-111.

[57] B.R. KONSYNSKI AND W.E. BRACKER. "Software Packages for Solving Network Puzzles" in *Data Network Design Strategies*, Ed. R. Sarch, McGraw Hill, New York, 1983, pp. 99-105.

[58] Z. KOZICKI. "An Approach to Computer-Aided Network Design", *IEEE International Conference on Communications*, June 1981, pp. 4.4.1-4.4.7.

[59] A. KUEHN AND M. HAMBURGER. "A Heuristic Program for Locating Warehouses", *Management Science*, vol. 9, July 1963, pp. 643-666.

[60] M. LESK AND T. MOUTON. "Credo-HG: A Graphical Tool for Data Network Planning", *Proceedings of the 9th International Conference on Computer Communication*, Tel Aviv Israel, October 1988, pp. 563-567.

[61] X. LIANG. "Method for Local Network Planning", *Proceedings of the 12th International Teletraffic Congress*, Torino, November 1988, pp. 563-567.

[62] G. LUGER AND W. STUBBLEFIELD. *Artificial Intelligence and the Design of Expert Systems*, The Benjamin/Cummings Publishing Company, Inc, Redwood City, California, 1989.

[63] L. MANTELMAN. "AI carves inroads: Network Design, Testing, and Management", *Data Communications*, July 1986, pp. 106-123.

[64] M.A. MARSAN, G. BALBO, ET AL. "TOPNET: A Tool for the Visual Simulation of Communication Networks", *IEEE Journal on Selected Areas in Communications*, vol. 8, no. 9, December 1990, pp. 1735-1747.

[65] J. MARTIN. *Systems Analysis for Data Transmission*, Prentice-Hall, Englewood Cliffs, New Jersey, 1972.

[66] M.M. MATSUBARA AND S. TILLI. "Problems and Solutions for Growing Packet Switched Network Traffic and Topology" *Proceedings of the 8th International Conference on Computer Communications*, September 1986, pp. 753-758.

[67] P. MCGREGOR. "Network Design: An Algorithm for the Access Facility Location Problem", *IEEE Transactions on Communications*, vol. COM-25, no. 1, January 1977, pp. 61-73.

[68] NETWORK MANAGEMENT INC. "Mind-Data", *product description*, 11242 Waples Mill Rd, Fairfax, Virginia, USA, 1989.

[69] T. NG AND D. HOANG. "Joint Optimization of Capacity and Flow Assignment in a Packet-Switched Communications Network", *IEEE Transactions on Communications*, vol. COM-35, no. 2, January 1987, pp. 202-209.

[70] T. NISHIDA, M. MURATA, ET AL. "PLANS: Modelling and simulation system for LAN", *Proceedings of the International Conference on Model. Techniq. Tools Perform. Anal.*, May 1984, pp. 235-258.

[71] L. PHELPS, network engineer, Netway Communications Ltd, *private correspondence*, August 1990.

[72] J. PREECE. "Some HCI Issues Concerned with Displaying Quantitative Information Graphically", *Proceedings of the 7th Interdisciplinary Workshop on Informatics and Psychology*, Austria, May 1988, pp. 209-225.

[73] M. RAHMAN AND M. LATIF. "Topological Design of a Distributed Computer Communication Network in a Noisy Environment", *Computer Communications*, vol. 12, no. 6, December 1989, pp. 331-336.

[74] M. REISER. "Performance Evaluation of Data Communication Systems", *Proceedings of the IEEE*, vol. 70, no. 2, February 1982, pp. 171-196.

[75] M. REISER. "A Queueing Network Analysis of Computer Communication Networks with Window Flow Control", *IEEE Transactions on Communications*, vol. COM-27, 1979, pp. 1199-1209.

[76] R. ROSNER. "A Tariff for Every Occasion" in *Packet Switching: Tomorrow's Communications Today*, Lifetime Learning Publications, Belmont California, 1982.

[77] V.R. SAKSENA. "Topological Analysis of Packet Networks", *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 8, October 1989, pp. 1243-1251.

[78] M. SCHWARTZ. *Computer-Communication Network Design and Analysis*, Prentice-Hall Inc, New York, 1977.

[79] M. SCHWARTZ. *Telecommunication Networks: Protocols, Modeling and Analysis*, Addison-Wesley, Reading, Massachusetts, 1987.

[80] K.S. SHANMUGAN. "An Update on Software Packages for Simulation of Communication Systems (Links)", *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 1, January 1988, pp. 5-12.

[81] R. SHARMA. *Network Topology Optimization: The Art and Science of Network Design*, Van Nostrand Rienhold, New York, 1990.

[82] R. SHARMA, P. DE SOUSA, ET AL. *Network Systems*, Van Nostrand Rienhold, New York, 1982.

[83] H. SMITH. "Using Graph Theory for Network Analysis", *Proceedings of the International Conference on Management and Performance Evaluation of Computer Systems*, December 1988, pp. 893-897.

[84] K.P. STEINRUCK AND G. GUTZMEROW. "Delay and Throughput Analysis for the German Packet Switched Public Data Network DATEX-P", *GLOBECOM '84*, 1984, pp. 25.2.1-25.2.4.

[85] SUN MICROSYSTEMS INC. "Pixrect Reference Manual (Revision A)", *product manual,* 1988.

[86] SUN MICROSYSTEMS INC. "SunView 1 Programmers Guide (Revision A)", *product manual,* 1988.

[87] SUN MICROSYSTEMS INC. "SunView 1 System Programmers Guide (Revision A)", *product manual,* 1988.

[88] M.R. TAIE, P.K. MEHROTRA, ET AL. "MUPLES: An Expert System for Multipoint Circuit Layout", *IEEE Global Telecommunications Conference & Exhibition*, Dallas Texas, November 1989, pp. 14.3.1-14.3.5.

[89] R. TARJAN. "Testing Graph Connectivity", *Proc. 6th Annual Symposium on Theory of Computing,* April 1974, pp.185-193.

[90] TELCO RESEARCH. Data Pathfinder System, *product description,* 1207 17th Avenue South, Nashville, Tennessee, USA, 1990.

[91] K. TERPLAN. "Network management evolution", *Computer Communications,* vol. 11, no. 5, October 1988, pp. 267-274.

[92] H. THIMBLEBY. *User Interface Design,* Addison-Wesley, Reading Massachusetts, 1990.

[93] F. TOBAGI, M. GERLA, ET AL. "Modeling and Measurement Techniques in Packet Communication Networks", *Proceedings of the IEEE,* vol. 66, no. 11, November 1978, pp. 1423-1447.

[94] H.J. VAN NORMAN, "A User's Guide to Network Design Tools", *Data Communications,* April 1988, pp. 115-133.

[95] R. VAN SLYKE, W. CHOU, ET AL. "Avoiding Simulation in Simulating Computer Communication Networks", *Proceedings of NCC,* 1973.

[96] VECTOR SOFTWARE INC. "Node/1 : Network Optimization Design and Engineering System", *product description,* 370 Lexington Ave, New York, USA, 1989.

[97] J. WANG AND E. WHITE. "Integrated Methodology for Supporting Packet Network Performance Management and Planning", *Computer Communications,* vol. 13, no. 9, November 1990, pp. 558-570.

[98] S. LAM AND C. HSIEH. "Modelling, Analysis and Optimal Routing of Flow-Controlled Communication Networks", *SIGCOMM '87,* 1987, pp. 162-172.

[99] K. CHANDY AND R. RUSSELL. "The design of multipoint linkages in a teleprocessing tree network", *IEEE Transactions on Computers,* vol. C-21, no. 10, 1972, pp.1062-1066.

[100] P. Clitherow and G. Fisher. "Knowledge Based Assistance of Genetic Search in Large Design Spaces", *2nd International Conference of Industrial and Engineering Applications of Artificial Intelligence and Expert Systems,* vol. 2, 1989, pp.729-733.